Call-A.P.P.L.E.™

World's Largest Apple User Group Magazine

May 1987 • $3.00 U.S.

Musicians' Tools

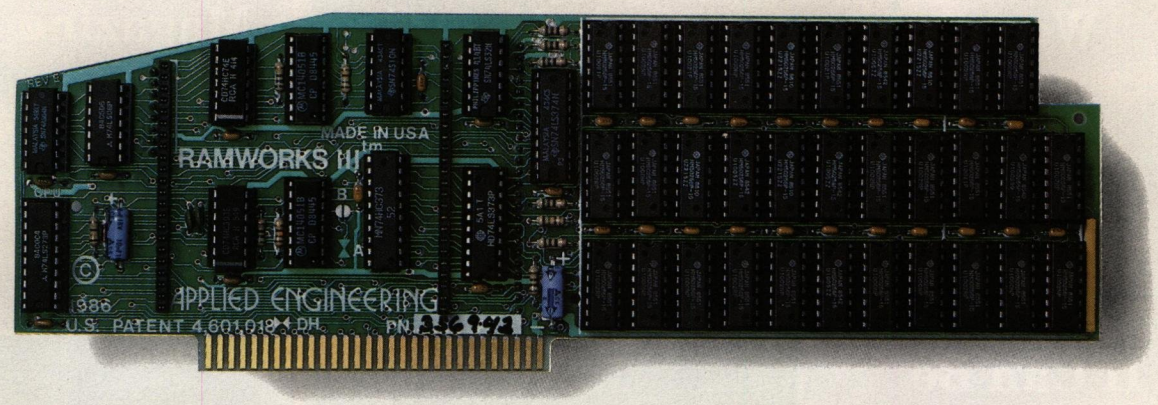
- You II Can Be Musical
- The IIGS Monitor
- Interrupt Card for the II
- Modular Assembly on the GS
- Cellular Automata Using TML Pascal
- Comparison of Three CAD Programs Part 2
- Apple II Panel from AppleWorld '87
- A.P.P.L.E. Doctor
- Write -A.P.P.L.E.

05
74470 19152 0

Apple PugetSound Program Library Exchange

RamWorks® III

Patented Performance from the Recognized Leader



With battery backed RAM port, RGB port, increased memory capacity, full software compatibility and more compact design, RamWorks III is a generation ahead.

RamWorks III is the newest 3rd generation RAM card for the Apple IIe. It incorporates all of the technology and improvements that years of experience and over a hundred thousand sales have given us. By selling more memory cards than anyone else and listening to our customers, we were able to design a memory card that has the ultimate in performance, quality, compatibility and ease of use. A design so advanced it's patented. We call it RamWorks III, you'll call it awesome!

The AppleWorks Amplifier.

While RamWorks III is recognized by all memory intensive programs, NO other expansion card comes close to offering the multitude of enhancements to AppleWorks that RamWorks III does. Naturally, you'd expect RamWorks III to expand the available desktop, after all Applied Engineering was a year ahead of everyone else *including Apple* in offering more than 55K, and we still provide the largest AppleWorks desktops available. But a larger desktop is just part of the story. Look at all the AppleWorks enhancements that even Apple's own card does not provide and *only* RamWorks III does. With a 256K or larger RamWorks III, *all* of AppleWorks (including printer routines) will automatically load itself into RAM dramatically increasing speed by eliminating the time required to access the program disk drive. Switch from word processing to spreadsheet to database at the speed of light with no wear on disk drives.

Only RamWorks eliminates AppleWorks' internal memory limits, increasing the maximum number of records available from 1,350 to over 25,000. *Only* RamWorks increases the number of lines permitted in the word processing mode from 2,250 to over 15,000. And *only* RamWorks offers a built-in printer buffer, so you won't have to wait for your printer to stop before returning to AppleWorks. RamWorks even expands the clipboard. And auto segments large files so they can be saved on two or more disks. You can even have Pinpoint or MacroWorks and your favorite spelling checker in RAM for instant response.

RamWorks, nothing comes close to enhancing AppleWorks so much.

The Most Friendly, Most Compatible Card Available.

Using RamWorks III couldn't be easier because it's compatible with more off-the-shelf software than any other RAM card. Popular programs like AppleWorks, Pinpoint, Catalyst, MouseDesk, Howard-Soft, FlashCalc, Pro-Filer, Managing Your Money, SuperCalc 3a, and MagiCalc to name a few (and *all* hardware add on's like ProFile and Sider hard disks). RamWorks is even compatible with software written for Apple cards. But unlike other cards, RamWorks plugs into the IIe auxiliary slot providing our super sharp 80 column text (U.S. Patent #4601081) in a completely integrated system while leaving expansion slots 1 through 7 available for other peripheral cards.

RamWorks III is compatible with all Apple IIe's, enhanced, unenhanced, American or European versions.

Highest Memory Expansion.

Applied Engineering has always offered the largest memory for the IIe and RamWorks III continues that tradition by expanding to 1 full MEG on the main card using standard RAMs, more than most will ever need (1 meg is about 500 pages of text)...but if you do ever need more than 1 MEG, RamWorks III has the widest selection of expander cards available. Additional 512K, 2 MEG, or 16 MEG cards just snap directly onto RamWorks III by plugging into the industry's only low profile (no slot 1 interference) fully decoded memory expansion connector. You can also choose non-volatile, power independent expanders allowing permanent storage for up to 20 years.

It Even Corrects Mistakes.

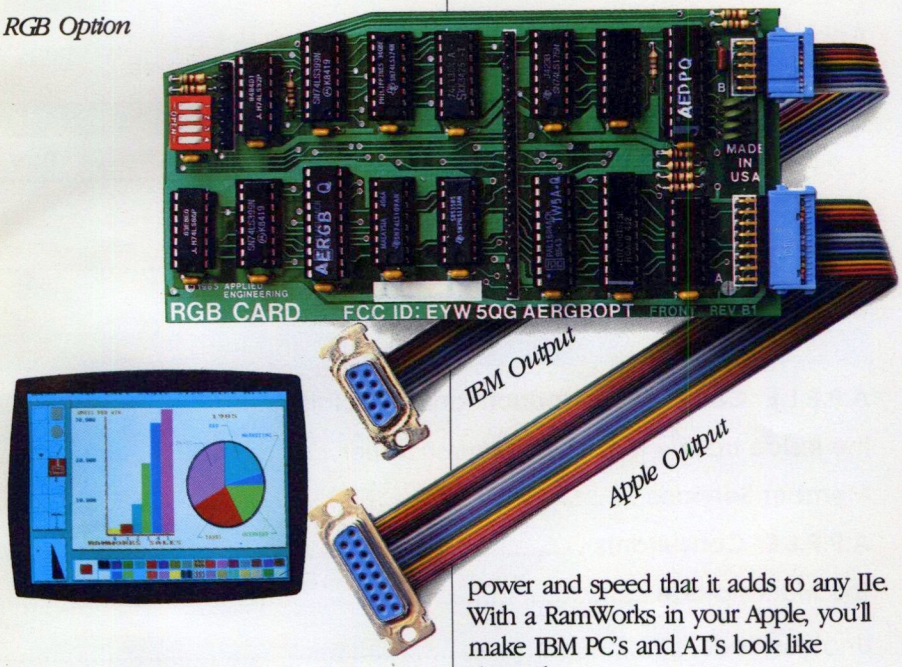
If you've got some other RAM card that's not being recognized by your programs, and you want RamWorks III, you're in luck. Because all you have to do is plug the memory chips from your current card into the expansion sockets on RamWorks to recapture most of your investment!

The Ultimate in RGB Color.

RGB color is an option on RamWorks and with good reason. Some others combine RGB color output with their memory cards, but that's unfair for those who don't need RGB *and* for those that do. Because if you don't need RGB

Applied Engineering doesn't make you buy it, but if you want RGB output you're in for a nice surprise because the RamWorks RGB option offers better color graphics plus a more readable 80 column text (that blows away any composite color monitor). For only $129 it can be added to RamWorks giving you a razor sharp, vivid brilliance that most claim is the best they have ever seen. You'll also appreciate the multiple text colors (others only have green) that come standard. But the RamWorks RGB option is more than just the ultimate in color output because unlike others, it's fully compatible with all the Apple standards for RGB output control, making it more compatible with off-the-shelf software. With its FCC certified design, you can use almost any RGB monitor because only the new RamWorks RGB option provides both Apple standard and IBM standard RGB outputs (cables included). The RGB option plugs into the back of RamWorks with no slot 1 interference and remember you can order the RGB option with your RamWorks or add it on at a later date.

RGB Option



True 65C816 16 Bit Power.

RamWorks III has a built-in 65C816 CPU port for direct connection to our optional 65C816 card. The only one capable of linearly addressing more than 1 meg of memory for power applications like running the Lotus 1-2-3™ compatible program, VIP Professional. Our 65C816 card does not use another slot but replaces the 65C02 yet maintains full 8 bit compatibility.

Endorsed by the Experts.

A+ magazine said "Applied Engineering's RamWorks is a boon to those who must use large files with AppleWorks...I like the product so much that I am buying one for my own system." inCider magazine said "RamWorks is the most powerful auxiliary slot memory card available for your IIe, and I rate it four stars...For my money, Applied Engineering's RamWorks is king of the hill."

"I wanted a memory card for my Apple that was fast, easy to use, and very compatible; so I bought RamWorks."

Steve Wozniak, the creator of Apple Computer

Apple experts everywhere are impressed by RamWorks's expandability, versatility, ease of use, and the sheer power and speed that it adds to any IIe. With a RamWorks in your Apple, you'll make IBM PC's and AT's look like slowpokes.

Quality and Support of the Industry Leader

RamWorks III is from Applied Engineering, the largest, most well supported manufacturer of Apple peripherals and the inventor of large RAM cards for the Apple. With our 5 year no hassle warranty and outstanding technical support, you're assured of the most trouble free product you can buy.

It's Got It All

- 15 day money back guarantee
- 5 year hassle free warranty
- Built-in super sharp 80 column display, (U.S. Patent #4601081)
- Expandable to 1 MEG on main card
- Expandable to 16 meg with expander cards, with NO slot 1 interference
- Can use 64K or 256K RAMs
- Powerful linear addressing 16 bit coprocessor port
- Automatic AppleWorks expansion up to 3017K desktop
- Accelerates AppleWorks
- Built-in AppleWorks printer buffer
- The only large RAM card that's 100% compatible with all IIe software
- RamDrive™ the ultimate disk emulation software included free
- Memory is easily partitioned allowing many programs in memory at once
- Compatible, RGB option featuring ultra high resolution color graphics and multiple text colors, with cables for both Apple and IBM type monitors
- Built-in self diagnostics software
- Lowest power consumption (U.S. Patent #4601081)
- Takes only one slot (auxiliary) even when fully expanded
- Socketed and user upgradeable
- Software industry standard
- Advanced Computer Aided Design
- Used by Apple Computer, Steve Wozniak and virtually all software companies
- Displays date and time on the AppleWorks screen with any PRO-DOS compatible clock
- Much, much more!

RamWorks III with 64K	**$179**
RamWorks III with 256K	**$199**
RamWorks III with 512K	**$239**
RamWorks III with 1 MEG	**$319**
RamWorks III with 1.5 MEG	**$489**
RamWorks III with 2 to 16 MEG	**CALL**
65C816 16 Bit Card	**$159**
RGB Option	**$129**
Optional Software:	
Pinpoint with RAM Enhancement Software	**$79**
VIP Professional	**$219**

RamWorks III. The industry standard for memory expansion of the Apple IIe. ORDER YOUR RamWorks III TODAY. See your dealer or call (214) 241-6060, 9 a.m. to 11 p.m., 7 days, or send check or money order to Applied Engineering.

MasterCard, Visa and C.O.D. welcome. Texas residents add 5⅛% sales tax. Add $10.00 if outside U.S.A.

Applied Engineering™
P.O. Box 798, Carrollton, TX 75006
(214) 241-6060

Call-A.P.P.L.E.

Volume 10, Number 5 May 1987

Official Publication of **A.P.P.L.E. Co-op** "People Helping People with Personal Technology"

About the Cover

The tools musicians use to create works of art vary— the soprano sax, the 4 string banjo... and now the II and IIGS. Cover photograph by Larry Neibauer.

Features

You II Can Be Musical • Marc Apfelstadt 10
A review of several music products for the Apple II and the GS.

Peeking at and Poking Around the Apple IIGs Monitor • Michael Fischer..15
A road map of what is in the monitor and some rules for your explorations.

A Simple Periodic Interrupt Card for the Apple II • Peter & Allen Baum.....22
Periodically 'wake up' your program without interfering with the main program.

Modular Assembly Pt. 5: The Apple IIGS • Bob Bishop..... 28
Modular assembly language guidelines for the GS.

Cellular Automata on the IIGS Using TML Pascal • Enrico Colombini 37
"Every tick of an imaginary clock, the automaton changes..."

Comparison of Three Cad Programs Pt. 2 • Lee Webster 42
The conclusion of this three product review, with a special focus on MATC-CAD.

AppleWorld '87 • Kathryn Hallgrimson Suther/Larry Neibauer..... 48
A special Apple II Panel discussion between eminent Apple employees.

Letters

A.P.P.L.E. Doctor • Bruce F. Field/Robert C. Platt..... 54
A nice array of hardware and software problems examined.

Write -A.P.P.L.E 56
Readers' Forum

Departments

A.P.P.L.E. Co-op New Products and Specials 5

the inside track • Kathryn Hallgrimson Suther 8

Member Services 9

A.P.P.L.E. Consultants 9

The Marketplace 59

Unclassifieds 66

Advertiser Index 63

Call -A.P.P.L.E. (ISSN 8755-4909) is published monthly by Apple PugetSound Program Library Exchange, 290 S.W. 43rd Street, Renton, Washington 98055. (206) 251-5222. Basic subscription price is $21 per year. Membership in A.P.P.L.E. Co-op, which includes magazine subscription, is $26 per year plus a one-time $23 application fee. Additional postage is required for First Class or foreign delivery.
Second-class postage paid at Renton, Washington, and at additional mailing offices. POSTMASTER: Send address changes to *Call -A.P.P.L.E.*, Address Change, 290 S.W. 43rd St., Renton, WA 98055.

A.P.P.L.E. Co-op New Products & Specials May 1987

Prices good through June 15, 1987

ProTERM by Greg Schaefer from Checkmate Technology, Inc. IIGS, IIe, IIc

Introducing possibly the easiest to use and most powerful telecommunications package to hit the Apple II market in years! ProTERM allows you to select from large lists of serial cards and modems for effortless no-hassle installation. Choose the IIGS by name and have instant installation on the internal serial port. The installation is just as easy on a IIe or IIc. All popular serial cards and internal/external 300,1200 and 2400 baud modems are supported with one-key selection.
ProTERM's AUTOLEARN feature creates powerful macros by recording your actions while you work. This capability allows you to record tedious log-on procedures and replay them with a press of a key. ProTERM supports all modern data transfer protocols including XMODEM, XMODEM (DOS/ProDOS), XMODEM CRC, YMODEM, KERMIT, and others. Other features include:

- On-line help screens available at any time.
- Unwanted characters filtered out automatically.
- Easy terminal emulation includes DEC VT-52, Televideo 950, Datamedia, Heathkit H-19, Hazeltine 1500, LSI ADM-3A.
- Batch sending mode allows you to transfer•entire disks including Operating systems!
- ProDOS clock support including ELAPSED TIME counter.
- Phone number list handler.
- Option to save captured text as lines, as paragraphs or single long piece.
- AUTOLEARN macro builder saves time and money.

Price: $79.00 | Code: **S2C1PT** | Wt: 1 LB. 3 OZ. |

ORCA/M For The Apple IIGS

ORCA/M is a complete development environment for assembly language programming on the Apple IIGS. The system includes the assembler with full macro capabilities, link editor, command processor, full screen text editor, and complete subroutine library support! The operating environment included with ORCA/M is fully compatible with the Apple Programmer's Workshop (APW). It provides a development environment for any compiler or accessory that is compatible with APW. Features include:

The Assembler
Assembles up to 5000 lines per minute.
Supports 6502,65C02, and 65816.
Supports partial assembly for even faster turn around of corrections.
Upwards compatible with ORCA/M 4.0 for eight bit Apple computers.
Full macro capabilities including:
- Macro libraries.
- Global communications between macros.
- Pass strings to macros.
- Macros can call macros.

The Command Processor
- UNIX-like I/O direction.
- Based on ProDOS 16.
- Advanced EXEC files with ability to Call other EXEC files, Pass parameters to EXEC files, Complete conditional execution, language making it possible to program the command processor.

The Editor
- Full screen text editor.
- Search and Replace.
- Cut, Copy, Paste and Delete blocks of text.

The Link Editor
- Automatic library searching.
- Create overlayed programs.
- Link multiple input files (for UNIX-style separate compilation).
- Symbol tables limited only by memory.

The Libraries
- Formatted input and output of numbers and strings.
- 2,4 and 8 byte integer math including: Add,subtract, multiply and divide, Square root and random number generation.

System Requirements

Apple IIGS or Apple IIe with a IIGS upgrade kit, 3.5" drive or hard disk, 256K of additional memory.
Price: $54.00 | Code: **SGB3OM** | Wt: 1 LB. 6 OZ. |

TML PASCAL For the Apple IIGS

TML Pascal has been designed to meet the needs of the broadest range of programmers possible for the Apple IIGS TML Pascal has numerous extensions for programmers accustomed to other Pascal implementations, thus achieving the greatest amount of compatibility possible.
Compatible with either the Apple Programmer's Workshop (APW) or ORCA/M 1.0 for the IIGS, TML Pascal supports units, random disk I/O, and standard subprograms such as Moveleft, Fillchar, etc. that are found in UCSD implementations of Pascal such as Apple Pascal. And of course language features from the Macintosh version of TML Pascal such as type casting, bit operations, CYCLE and LEAVE statements, Unit bodies, and much more are found in TML Pascal for the Apple IIGS
- Full access to entire Toolbox.
- Runs in full 16-bit native mode under ProDOS 16 and APW (ORCA/M).
- Allows you to develop:Stand-alone ProDOS16 applications, New Desk Accessories, APW tools, Create definition procedures.

System Requirements
APW or ORCA/M 1.0 for the IIGS, One 3.5" 800K floppy,
Memory Expansion for a total of 768K, for development of large applications, two 800K floppies or a hard drive is recommended.
Price: $99.00 | Code: **SGTAPW** | Wt: 13 OZ. |
(See page 37 for an article implementing TML.)

MUSIC CONSTRUCTION SET

from Electronic Arts
- **Apple IIGS compatible!**
- SUPERSONIC compatible for stereo output.
- MIDI compatible.
- 16 Digitized instruments.
- Supports Imagewriter printers.

Price: $31.00 | Code: SGE1MC | Wt: 6 OZ. |
(See page 10 for review article.)

APPLE IIGS TECHNICAL REFERENCE

by Michael Fischer
Everything you need to know about the Apple IIGS.
Price: $16.95 | Code: **SGE1MC** | Wt: 2 LBS. 5 OZ. |

ORDERING CHECKLIST

To speed all orders, please include:

1. Your Name as it appears in our records (same as on your mailing label).
2. Member number and expiration date.
3. Complete STREET ADDRESS (**not** Box number — see explanation below).
4. Phone number to reach you by day if we have questions.
5. Payment Including:
   Handling charge if applicable
   Domestic or Foreign Shipping Charges
   Magazine Postage if applicable
   (Sorry we do not accept COD or Purchase Orders on individual memberships.)

Each item you order needs:

6. Complete product name (and version if there is more than one product of that type). Specify computer model you are using.
7. Product Code

SHIPPING ADDRESS: We normally ship products by United Parcel Service (UPS). Therefore, the shipping address on your order cannot be a post office box. You may specify a P.O. Box for orders weighing less than 8 ounces. You may also make special arrangements for shipping some items via US Mail, Air Parcel Post, or UPS Second Day Air.

DOMESTIC SHIPPING CHARGES

SHIPPING WEIGHT	COST
Up to 5 Lbs.	$ 2.00
5-10 Lbs.	3.00
10-15 Lbs.	4.00
15-20 Lbs.	5.00
20-25 Lbs.	6.00
25-30 Lbs.	7.00
30-35 Lbs.	8.00
35-40 Lbs.	9.00
40-45 Lbs.	10.00
45-50 Lbs.	11.00

UPS 2ND DAY AIR

WEIGHT NOT TO EXCEED	48STATES A	HAWAII D
1 lb.	$ 3.00	$ 4.53
2 "	4.00	5.71
3 "	5.00	6.89
4 "	6.00	8.07
5 "	6.50	9.25
6 "	7.50	10.43
7 "	8.50	11.61
8 "	9.50	12.79
9 "	10.50	13.97
10 "	11.50	15.15
11 "	12.50	16.33
12 "	13.50	17.51
13 "	14.50	18.69
14 "	15.50	19.87
15 "	16.50	21.05
16 "	17.50	22.23
17 "	18.50	23.41
18 "	19.50	24.59
19 "	20.00	25.77
20 "	21.00	26.95
21 "	22.00	28.13
22 "	23.00	29.31
23 "	24.00	30.49
24 "	25.00	31.67
25 "	26.00	32.85

FOREIGN SHIPPING CHART

POSTAL TABLE	Zone I West Indies Central America, Columbia, Venezuela	Zone II All South American Countries (except in Table I), Europe, N. Africa	Zone III Asia, Mid East, S. Africa Australia and Pacific Islands
0-4 oz.	$ 1.51	$ 1.96	$ 2.41
4-8 oz.	2.13	3.95	3.95
8-12 oz.	2.75	4.12	5.49
12-16	3.37	5.20	7.03
16-20 oz.	3.99	6.28	8.57
20-24 oz.	4.61	7.36	10.11
24-32 oz.	5.85	9.52	13.19
2-3 lbs.	8.29	13.84	19.39
3-4 lbs.	10.73	18.16	25.59
Each additional ½ lb. over 4 pounds add:	1.22	2.16	3.10

CANADA SHIPPING RATES

0-4 oz.	$ .76
4-8 oz.	1.48
8-12 oz.	2.20
12-16oz.	2.84
2 lb.	3.92
3 lb.	5.00
4 lb.	6.08

For weights over 4lbs: $6.08 + $1.08 additional per pound.

ORDERING

ORDER TOLL FREE
1-800-426-3667
(Outside Washington State)

Date _____

Name _____ Membership # _____ Expires _____

Address _____

City _____ State/Country _____

Zip _____ Phone # ( ) _____

Soc. Sec./Tax I.D. # (for dividend payments) _____

☐ Check enclosed ☐ VISA ☐ MasterCard

Account # _____ Exp. Date _____

	CODE	QTY.	PRODUCT	Shipping WT.	Unit Price	Amount
1.						
2.						
3.						
4.						
5.						
6.						

Total Shipping Wt. _____ Sub Total _____

*Non-Member add 10% _____

Washington Residents add 8.1% Sales Tax _____

TOTAL SHIPPING COST _____

$3.00 Handling Charge If Order Is Under $25.00 _____

TOTAL AMOUNT _____

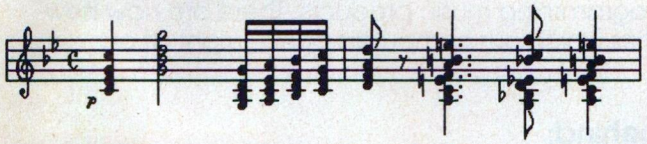
Mail to: A.P.P.L.E. Co-op
290 S.W. 43rd St.
Renton, WA 98055
(206) 251-5222

The Leading Music Notation Tool for all Apple II® Systems.

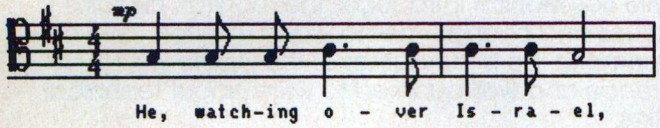
MusicPrinter™ writes, edits and prints complete music scores.

New *MusicPrinter* lets you create, edit, transpose and print complete music scores. *MusicPrinter* is the most advanced word processor for music—notes, stems, flags, beams, accidentals and text can be placed exactly where you wish. Make changes, edit and re-edit—your changes appear instantly, on screen, as you work.

Write chords



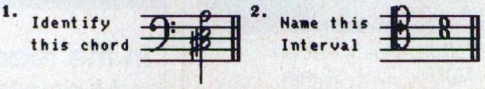
Add text



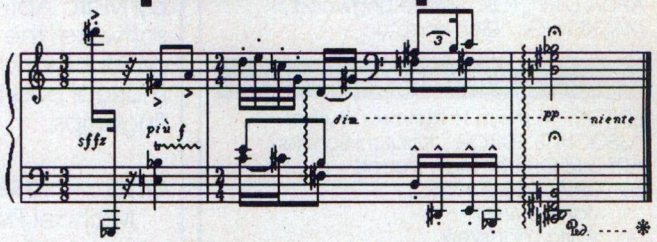
Full scoring

MusicPrinter produces complete multi-instrument scores and multi-voice choral scores. Complete individual parts can be pulled automatically from the full score.

Create tests



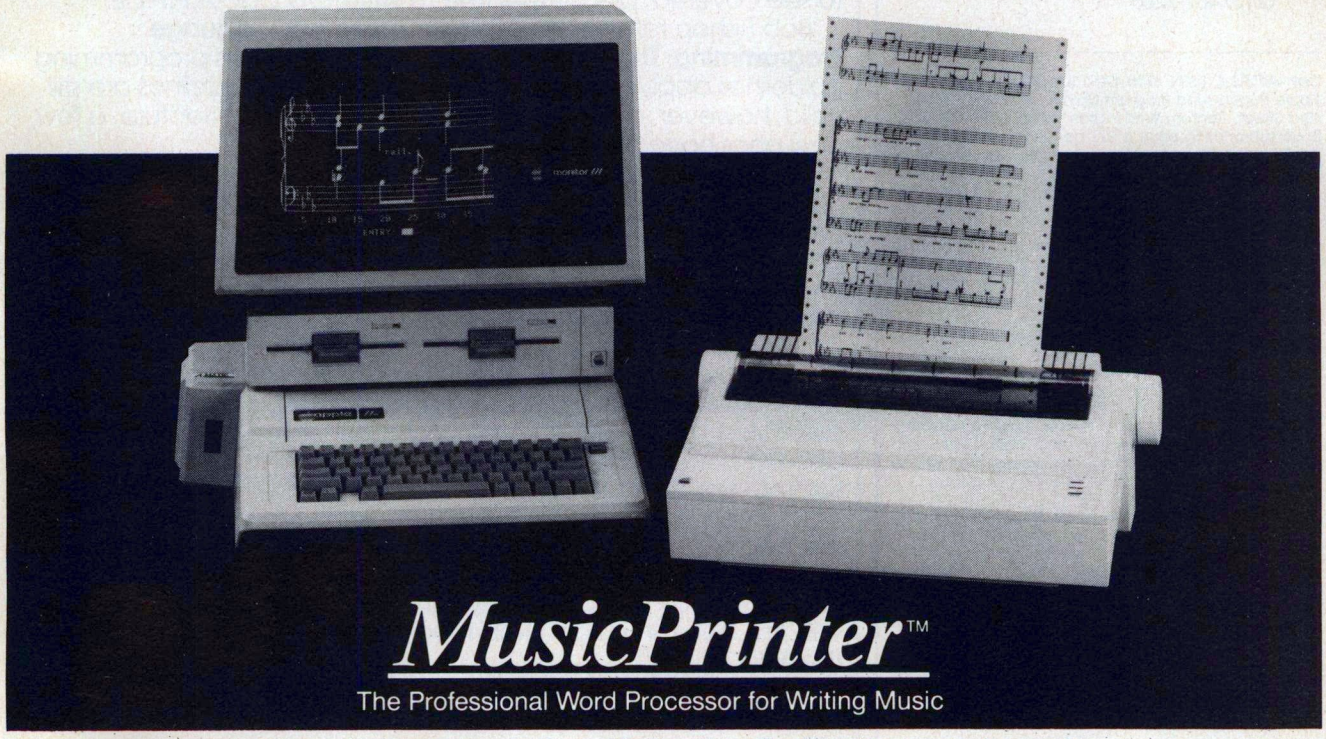
Complex is simple



MusicPrinter writes music faster than a skilled copyist. You can quickly make a complete score that is performance-ready. *MusicPrinter* provides a complete set of special music characters as well as three lettering styles.

How to order

MusicPrinter is priced at $149 plus shipping and works with all Apple II® computers, most printers and interface cards. MusicPrinter can support a mouse or joystick for extra input speed. To preview or order call **(800) 426-2673.** Washington customers, please call (206) 462-1007. VISA or MasterCard welcome. Money-back guaranteed.



MusicPrinter™

The Professional Word Processor for Writing Music

Call or write for our new catalog—50 *MicroMusic*™ software programs that teach!
Temporal Acuity Products, Inc., 300 - 120th Ave. N.E., Building 1, Bellevue, WA 98005 **(206) 462-1007.**
MusicPrinter designed exclusively for Temporal Acuity Products by Jack M. Jarrett.

Apple II® is a trademark of Apple Computer, Inc.

CIRCLE 14 ON READER SERVICE CARD

A.P.P.L.E. Co-op

OFFICERS AND DIRECTORS

PRESIDENT: Bob Huelsdonk
VICE PRESIDENT: Merle H. Davis
TREASURER: Dave DeGroot
SECRETARY: David F. VanBerkem
DIRECTORS: Michael D. Branham
Jack Connick
Don Williams

ADMINISTRATION

EXECUTIVE DIRECTOR: Dick Hubert
GENERAL MANAGER: Don Steele
PRODUCT DEVELOPMENT/ EDUCATION MANAGER: Charles Stillman
APDA DIRECTOR: Dave Lingwood
MARKETING: Rich Knapton

Call -A.P.P.L.E.

EDITOR: Kathryn Hallgrimson Suther
ASSOCIATE EDITOR: Kaiulani Schuler
ART DIRECTOR: Mary Hubert
EDITORIAL ASSISTANTS:
Janine West
Karla Landsverk

TECHNICAL CONSULTANTS:
Walt Gardner
Larry Neibauer
Keith Nemitz
Charles Stillman

ADVERTISING

NATIONAL COORDINATOR:
John Elliott (206) 251-5222
CALIFORNIA REPRESENTATIVE:
JE Publishers' Representative Co.
6855 Santa Monica Blvd., Suite 200
Los Angeles, CA 90038
(213) 467-2266

Call -A.P.P.L.E. (ISSN 8755-4909) is published monthly by Apple PugetSound Program Library Exchange, 290 S.W. 43rd Street, Renton, Washington 98055, (206) 251-5222. Basic subscription price is $21 per year. Membership in A.P.P.L.E. Co-op, which includes magazine subscription, is $26 per year plus a one-time $23 application fee. Additional postage is required for First Class or foreign delivery.

Second-class postage paid at Renton, Washington, and at additional mailing offices. POSTMASTER: Send address changes to *Call -A.P.P.L.E.*, Address Change, 290 S.W. 43rd St., Renton, Washington 98055.

Entire contents copyright © 1987 by Apple PugetSound Program Library Exchange. All rights reserved. Authors and advertisers are responsible for the accuracy of their submitted material. Opinions expressed do not necessarily represent those of A.P.P.L.E. Co-op. Apple users groups may reprint editorial matter contained herein for non-commercial purposes, provided author, title, and publication credits are given. The address and phone number of *Call -A.P.P.L.E.* should also be included.

A.P.P.L.E. Co-op is an independent support group of Apple product users, and is not affiliated with Apple Computer, Inc. The following are trademarks of Apple Computer, Inc.: Apple, Apple II, Apple II+, Apple IIe, Apple IIc, Apple IIgs, Macintosh, Macintosh Plus, Macintosh XL, Lisa, ImageWriter, LaserWriter Plus, MacPaint, MacWrite, DOS 3.3, ProDOS, AppleWriter, AppleWorks, Applesoft, and UniDisk 3.5.

Printed in U.S.A.

the inside track

Kathryn Hallgrimson Suther

in the lead:

Musicians today, whether amateur or professional, can use Apples to help with musical explorations. **You II Can Be Musical**, by Marc Apfelstadt reviews three different types of music software: the "Enter, Play, Print" variety, a music typesetting product, and programming music products. There are now new musical possibilities which our computers offer — right at our fingertips.

and... close behind:

Michael Fischer joins us this month, with his first, in a two part series of articles on the Apple IIGS monitor. **Peeking at and Poking Around the Apple IIGS Monitor** introduces the monitor, explains how to enter values and change memory and registers, and how to use the disassembler. He is the author of the *Apple IIGS Technical Reference* and well qualified to write about technical aspects of the IIGS.

A Simple Periodic Interrupt Card for the Apple II describes the Jeeves card Peter and Allen Baum created, and the subroutine which implements an alarm clock and checks for keystrokes. If you are interested in interrupts, this one's for you. Also, Peter and Allen wish to remind you that they are interested in your article ideas. If there is a hardware subject you would like to see covered, please feel free to write c/o *Call -A.P.P.L.E.*

Bob Bishop returns with **Modular Assembly Language Programming: The Apple IIGS** and explains how his programming guidelines apply to this new computer. All the guidelines are still valid, however, due to the GS's complicated architecture, a few new ones have been added.

Cellular Automata on the IIGS by Enrico Colombini uses TML Pascal to display linear cellular automata on the color screen. This program displays every new generation with a color *under* the previous one, so we can see the "history" of the automaton through generations. By connecting the borders together, Enrico creates a "circular automaton".

In March, Lee Webster reviewed two CAD programs, *discoverCAD and CADDRAW*. This month he concludes his **Comparison of Three CAD Programs** by looking at *MATC-CAD*, a product by Milwaukee Area Technical College. He also discusses the documentation for all three products and summarizes their features.

Thanks to Larry Neibauer, we have included a transcription of the Apple II Panel which took place at **AppleWorld '87** the first week in March. This discussion was moderated by Tom Virden, and included input by Harvey Lehtman, Neil Amsden, Rich Williams, Mike Smith and Rob Moore — all well known experts in the field.

We feel there is something for everyone in this issue.

II

MEMBER BILLBOARD

Election

Here are the results of the election:

President:	Bob Huelsdonk
Vice President:	Merle H. Davis
Treasurer:	Dave DeGroot
Secretary:	David F. VanBerkem
Directors:	Michael D. Branham
	Jack Connick
	Don Williams

These men bring the experience and leadership that will be needed to help chart the course of the co-op in the coming year.

Member # 50,000

On February 23, 1987, Mohamed M. Zamir of the Bronx, NY became member number 50,000. We are sending Mr. Zamir a letter notifying him that he has been awarded $100 credit which may be applied to the purchase of any A.P.P.L.E. Co-op product. Congradulations Mr. Zamir.

Computer Fair

We have been asked by the Pacific Science Center here in Seattle to be in charge of the 1987 computer fair. This is a great opportunity for the co-op to expand its membership base, and to receive valuable recognition. However, in order for this type of thing to succeed, we need the voluntary involvement of the co-op members. If any of you would like to work on such a project, or have experience in this type of endeavor, please call one of the officers of the co-op.

//

MEMBER SERVICES

MAIN OFFICE: (206) 251-5222 • 8:30 a.m. - 4:00 p.m. PST • Customer Services **(206) 251-9798** • Order Desk • Membership Information • Accounting • Dealer Services • Marketing • Testing and Evaluation • Publications • Education

24 Hour Toll-Free Order Line: 1-800-426-3667 (U.S. except Washington state), **1-800-527-7562** (Within Washington State), **1-800-237-4644** (Canada). ORDERS ONLY, INFORMATION NOT AVAILABLE.

Retail Store: (206) 251-6787 • M-F 10:30 a.m. - 6:00 p.m. Sat. 10-5 (PST) Store-related calls only, please.

Programming/Technical Assistance: (206) 251-0071 Call -A.P.P.L.E. HOTLINE • 9:00 a.m. - 3:00 p.m. PST Weekdays.

A.P.P.L.E. Crate Bulletin Board: Wes Munro, sysop (206) 251-0543.

On-line Catalog
GEnie, American Apple Round Table Bulletin Board, category 7. Information call 1-800-638-9636.

Meetings: • 7:30 p.m. • 290 S.W. 43rd St., Renton, WA 98055 (except as noted).• **Co-op:** 3rd Tuesday of month • **Mac SIG:** 2nd Tuesday of each month • **Educators SIG:** 7:00 p.m., 4th Tues. of month • **Amiga SIG:** 2nd Wed. of month • **MS DOS SIG:** 4th Thursday of month.

Membership Information: available on postpaid tear-off card inside.

A.P.P.L.E. CONSULTANTS

Call -A.P.P.L.E. consultants are unpaid volunteers and perform a valuable service. Before placing a call, please make sure you know the time zone you are calling. No calls after 9pm please. Sometimes a consultant will have you leave your name and number and later return your calls. In all fairness, please expect the return call to be collect. We are always looking for more consultants. If you see an area that needs coverage, please drop us a note c/o *Charlie Stillman* with your qualifications and availability. The codes to the right of the phone numbers refer to time zones:

P—Pacific	M—Mountain	C—Central	E—Eastern
ie.: 1:00	2:00	3:00	4:00

'C'	PhilPearson	(603) 362-4297	(E)
Accounting (East)	Robert Greco	(609) 884-3047	(E)
Agriculture	Neil Clemmons	(314) 968-9618	(C)
Apple III(Belgium)	Van der Straeten	(015) 205328	
Appiesoft	Leon Raesly	days(301) 439-1799	(E)
Appiesoft (Austrailia)	Graeme Samways	048-851-541	
AppleWorks	Ina Levinson	days(713) 524-6747	(C)
AppleWorks (East)	Eugene Whitehouse	(201) 434-5110	(E)
AppleWorks (Norway)	Dennis Dahl	04-674-248	Ap
pleWorks(India)	Arun Nath	660060 ext. 609	
AppleWorks	Leon Raesley	days(301) 439-1799	(E)
AppleWorks/Business (Israel)	Jeanni Spindel	972-052-83732	
Apple IIc	Leon Raesley's	(301) 439-1799	(E)
Apple IIc (MidEast)	Paul Rich	(Qatar) 418822	
Apple IIe (India)	Sanjay Kalra	660060 ext. 610	
Apple III (East)	Carmen Picardi	(603) 895-4447	(E)
Apple III (East)	Harry T. Hanson	(201) 467-0712	(E)
Apple III (East)	Stuart Yusem	(609) 822-9306	(E)
Apple III (West)	Bob Huelsdonk	(206) 542-7122	(P)
Apple III Pascal	Dennis R. Cohen	(818) 956-8559	(P)
Apple IIe (East)	John Hall	(617) 241-9050	(E)
Apple IIGS	Marc Apfelstadt	(919) 282-1494	(E)
Apple IIGS	Cecil Fretwell	eves(319) 236-0961	(C)
AppleWriter	Leon Raesley	days(301) 439-1799	(E)
AppleWriter	Don Lancaster	(602) 428-4073	(M)
AppleWriter II	Leon Raesley	days(301) 439-1799	(E)
Assembly Language	Don Lancaster	(602) 428-4073	(M)
Assembly Language	Scott Zimmerman	(801) 378-4100	(M)
Assembly/CP/M(East)	Dennis Baumgarte	(914) 856-2721	(E)
Assembly/ProDOS	Sean Nolan	(617) 547-1723	(E)
CP/M(PCPI)	Doug Trueman	(417) 679-3526	(C)
CP/M & Wordstar	Leon Raesley	days(301)-439-1799	(E)
CP/M & Wordstar (Southern Africa)	Philip Christensen	(Maseru)311056	
DBASE	Leon Raesly	days(301) 439-1799	(E)
DBASE II (India)	Kalpana Dube	660060 ext. 610	
DOS (South)	Lee Meador	(817) 469-6019	(C)
DOS (West)	William Reynolds	(503)393-6758	(P)
DOS/AppleSOFT Europe	Cornelis Bongers	31-010525511ext.3275	
Education	Ted Perry	(916) 485-1690	(P)
Education Malaysia	Anthony Chan	(06) 711645	
Fortran	Donald Geenen	(414) 734-3904	(C)
Franklin	Doug Trueman	(417) 679-3526	(C)
Franklin (MidWest)	Glenn Hayhurst	(303) 388-7409	
Franklin (MidWest)	Doug Trueman	(402) 379-3006	
Geoscience	Tedd Sperling	(517) 393-1296	(E)
Handicapped Applications	Tom Deal M.D.	(206) 634-2847	(P)
Hardware/Am-Euro compatibility	Dennis Dahl	(Norway)04-674-248	
Hardware Design, I/O Control	Stan Axelrod	(415) 563-1840	(P)
Hardware Interfacing	Randy Halvorson	(206) 783-9040	(P)
Hardware Interfacing/Control (East)	Maurice Khano	(404) 924-1371	(E)
Hardware (Australia)	Graeme Samways	048-851-541	
Hi-Res	Ed So	(408) 267-3448	(P)
LaserWriter/Postscript	Don Lancaster	(602) 428-4073	(M)
Lisa	Fred Reitberger	(617) 587-9829	(E)
LISP	Frederick Naef	(703) 471-1479	(E)
Logo	Jeff Sandys	(206) 784-3317	(P)

CONTINUED ON PAGE 65

You II Can Be Musical!

Marc Apfelstadt

Musicians and Apples have a way of working together. With some newly released software, the flexibility and power of that relationship has expanded exponentially. This article will take a look at some of the current products and their applications for professional, amateur, and fun-loving musicians.

As I see it, there are three basic types of music software. The first is the "Enter, Play, and Print" species. The second possible application would be "Music Typesetting", and the third, "Programming Utilities". Each of the three areas has new software available which greatly increases the user's capabilities.

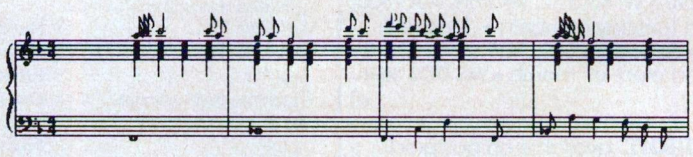
With the release of two new products for the IIGS, "Enter, Play, and Print" software has taken a quantum leap in the power available to the home user. These products also show how a basic concept can have widely varied implementation on screen.

Enter, Play, and Print

Music Electronic Arts' Construction Set (IIGS) takes advantage of the IIGS sound and graphics capabilities. Fifteen simultaneous voices can be used, along with the familiar MCS entry and editing screen. In addition, the program comes with a host of pre-defined timbres, or tone colors. Any timbre can be assigned to either the treble or bass clef, and remains in effect for the entire song. Stereo sound is also supported, with each clef going to a separate speaker when run through the appropriate hardware. As notes of similar rhythm stack vertically on the displayed staff, the stems automatically combine (two simultaneous eighth, notes, for example). This gives the MCS printed output an edge over that generated from The Music Studio. When notes occur near the upper screen limit, stems and flags can be cut off in printing. There is not a "beaming" feature, either... in 'Printed' music, two eighth notes often share a common flag. Still, as Figure 1 shows, the output is quite legible and acceptable for most purposes. If you have upgraded from a IIe and kept your old parallel printer, you might get a surprise with MCS. There is a basic assumption that you have an ImageWriter II if you have a IIGS, and it is not possible to define a printer configuration. A similar problem exists with The Music Studio.

MCS supports MIDI output for playback. With the proper board installed, Musical Instrument Digital Interface communication allows your computer to control playing of examples on external synthesizers and other digital instruments. At this point, the Passport MIDI board is the only one which seems to work... the Roland

FIGURE 1



board & software are not IIGS compatible in the current release.

Music Construction Set (GS) is built solely around efficient music entry and playback. There isn't any option to add lyrics (words) to your music, and the built-in timbre files cannot be altered. If you wish to have streamlined music entry and playback combined with above average print quality, MCS is the strongest contender.

The Music Studio, from Activision, takes an entirely different slant at the same goal. In many ways, this program puts digital recording studio techniques within the reach of the home user in an unusual way. To begin with, 'Color' is a basic premise of the system. Not only are 15 voices supported, any one song can use up to 15 individual instrument timbres. Each 'Tone' color is given a different 'Screen' color, so that Alto Sax notes are green, String Bass notes, brown, etc. In playing a scrolling score, the instrument sounds have 'Visual' reinforcement.

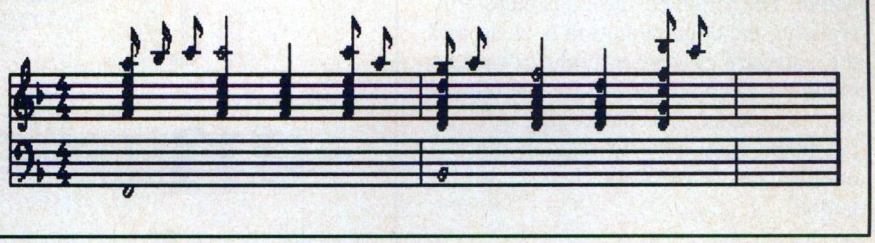
There is also a sound 'Editor', so that you can visually modify and create your own sound timbres by adjusting the attack, delay, volume curve, and other factors. The process works best with instruments where the fundamental (primary) pitch is dominant, like drums and strings. Instruments with a lot of harmonic overtones, or upper sound frequencies, are difficult to emulate (bassoon and flute, for example). The disk comes with classical, rock, jazz, and latin 'wave banks', as they are called, and the demo songs that use them are most impressive and inspirational.

The software also allows you to fake a four track recorder, assigning instruments to tracks by color. In that way, you can play the example, without certain instrument groups, simply by turning off the track indicator they are assigned to... omitting the drums from a latin tune in order to check the bass line, for instance.

MIDI is supported in The Music Studio, as is stereo sound — both with the proper hardware, of course. Interestingly enough, one feature of the sound editor is that each instrument can be individually assigned to either the left or right channel. The 'Macintosh' graphic concept is very obvious in this product, with pictures representing the various editing functions (insert, repeat, etc.,) throughout.

There are just a few minor drawbacks to the implementation. Where most software products have menus which pull down from the top, The Music Studio has too many options! A couple of toggles on the bottom of the display, for key and time signature, also trigger menus that roll down for selection. They work fine, but it is a bit disconcerting at first. The upper right corner has the 'Edit' options (graphics for repeat, insert, etc.), and an 'Options' choice above that on the top line for MIDI, Sound Design, and some other system level choices. I must say, though... once you've played around the screen for a few days, these quirks fade into insignificance beside the raw musical power at your disposal.

FIGURE 2



The quality of the printed output from The Music Studio is a more severe problem. Printing is very slow in this release, and the characters are blocky and roughly formed, as shown in Figure 2. If you need a combination of sound output and quality print, lean towards the Music Construction Set. If the *sound* is your primary motive, The Music Studio has it hands down.

Both of these products work with the internal speaker of the IIGS. For maximum enjoyment, consider picking up a *Supersonic Stereo Card* from MDIdeas. This add-on board is very easy to install... simply plug it into a slot, hook on a connector to the motherboard, and close the hood! Software that produces compatible stereo signals will automatically be split, and that which doesn't, will have standard mono output passed through. In my IIGS, I unplugged the connection for the built-in speaker to avoid adding to the sonic confusion, though I miss the occasional error beep. The SuperSonic Stereo Card performs exactly as advertised, and is low in price, considering the quality of sound enhancement. If you plan to use music software, this card is a must.

The board seems to use the slot only for power connections... you can still have the Modem port selected, for example. I'm a little curious as to the software protocol in that case, too, since my second UniDisk is SmartPort mapped to Slot 2, the Modem is selected from the Control Panel, and the SuperSonic Stereo board physically sits there! No problems yet, although I haven't tried to communicate online under this configuration.

As if the Stereo capability weren't enough, MDIdeas also sells a reasonably priced *SuperSonic Digitizer* which plugs onto an edge connector on the Stereo board. That device takes audio input, converts it into digital sound in the computer's memory, and allows the sound to be replayed through the Ensoniq chip.

There are a few things to be aware of regarding the digitizing board. First, the demo program requires a minimum of 1 meg RAM card. At the sampling speed of the demo software, 1 meg holds about 32 seconds of mono sound. When I hear people ask what a 16 meg RAM card will be good for in a GS, I'm tempted to reply "about 2 minutes of stereo music!" In order to use the Digitizer, you need to set the Control Panel to 'Your Card', and the demo software requires that Slot 2 be used.

The Digitizer is easy to install, and it performs its functions very efficiently. As a matter of fact, since it is using a digital recording process, the sound is almost *too* good. Recording an example from tape picks up all of the tape hiss, and live recording includes little details like IIGS *fan* hum. For best results, consider recording a Compact Disc or radio broadcast of a CD performance. The demo software is functional, but MDIdeas is leaving it to others to release polished editing software. For the price, the functions are very useful.

There is one other digitizing product in the pipeline that is also worthy of mention. *Future Sound,* from Applied Visions, is being ported over from the Amiga environment. The software has many functions of a tape recorder (rewind, fast forward, etc.), and it digitizes sound input for later replay. The product will also include a custom analog to digital converter board to handle the sound input process. If the Amiga version is paralleled closely, recorded sounds will be available for your use under program control!

At this point, I've seen a very im-

pressive demo, just beyond the alpha stage, with most of the software features active. The digitized speech is incredibly realistic, the documentation is clear, and the software is very straightforward to operate... keep an eye out for the release of this product. Of course, *that* A/D input board may have to park in the slot next to your SuperSonic Stereo card for output!

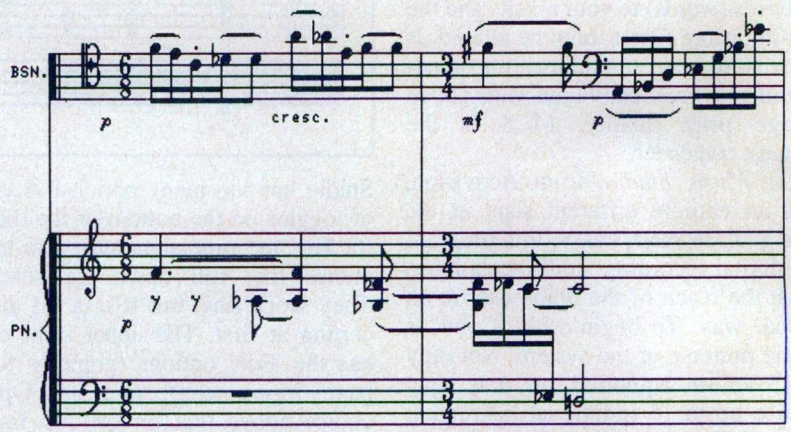
Music Typesetting

Music typesetting is the second potential application of Apples to music. One reason for a separate category is that the "Play & Print" programs are built around a grand staff, or piano score, which always contains both treble and bass clefs. There are other compromises in the combination products, mostly arising from the playing process, which prevent them from serving as a professional level of music typesetting program. Fortunately, there is a nonplaying alternative with graphic strength.

Music Printer, from Temporal Acuity Products, is a very specialized product. There is not any attempt at incorporating a 'Play' feature... rather, the software provides absolute control over the 'Printed' output generated. In a sense, the program functions as a 'musical typewriter'. You work on one line or grand staff at a time, scrolling across the page between left and right margins. Entry of notation is through mnemonic keyboard commands... QU for 'quarter, stem up', HD for 'half, stem down', etc. A command chart is provided showing all of the graphic symbols and their codes, and they are logical enough to become second nature quite easily. In addition to the full music character set, there are also three typefaces available for text on the score.

The new release of the program, version 2.0, offers several enhancements over the original. The IIGS is now supported, and a mouse or joystick can be used to place symbols on the screen. Beaming is supported between notes of similar stem direction, and the graphic character resolution is absolutely superb on the printed page. Print options include multiple passes up to nine times, and you can extend bar lines above and below the staff. What that means, of course, is that you can prepare a true conductor's score of music, and not be limited to a piano staff (grand staff... treble and bass). Figure 3 shows the print quality and flexibility available. Compare the output with that from the "Enter & Play" programs, and you'll quickly see the difference a dedicated print function makes.

FIGURE3
Sonata for Bassoon + Piano
Oskar Morawetz



The program also has block copy and transposition features, which adjust chromatics (flats, etc.) when moving passages. Any drawbacks? Well, just a couple. First, the copy protected disk comes without backup (like MCS & The Music Studio). TAP has an excellent record for turn-around of replacement disks, but it is a consideration to keep in mind with each of these products. The current version is supplied on 5 1/4 floppy running under DOS 3.3, too... no 3.5 format available. For beamed passages, Music Printer provides straight beams only... no slanted ones. That wrinkle is really a function of the resolution of dot matrix printers, since only a plotter can handle that type of drawing accurately. These are mostly "nit-picking minor" details, and they don't reduce the power or flexibility of the program in any way.

If you own Music Printer version 1.0, contact TAP about an upgrade. For the machine language programmers among the readers, you might be interested to know that the program was written by Jack Jarrett, who has a background in college music teaching. The source code is about 9000 lines of assembly language!

I have found Music Printer to be an excellent tool for producing high quality music output. Especially when you use a copy machine to reduce the examples slightly, the resolution is very satisfactory, and almost indistinguishable from medium quality printed scores. In my teaching work, the program has been used to print musical examples right in the middle of exam formats. I can recommend it highly where output quality is critical... it really has enough power to permit home music publishing.

Programming Music

Programming music is the third area where a computer can be utilized, and it is one which has posed an unusual problem. You see, hardware manufacturers are interested in selling music boards... so they focus on releasing software which runs *on their board*. Compatibility has always been a major problem in music software... until now!

What would you think of a music programming tool which runs under DOS 3.3 and ProDOS, and is easily available to both BASIC and Assembly Language? How about using a 'driver' concept for the sound production, so that *one* music code file can be played on a variety of hardware setups? Released on a disk which is not copy protected?

The Musician's Toolbox, from Wm. C. Brown, is all of that and more for $39.95 — *and* you can use the routines in your own published software with just a few minor stipulations. This newly released package could easily be the cause of a grassroots explosion in music programming. Let's look at the features, and you'll see what I mean.

The software is designed as a package of special routines for displaying music and text on the hi-res screen, and playing through a variety of sound sources. Since the 'program hooks' and music code are consistent, your program just needs to load the right 'sound play-

ing driver' at startup. Music can then be played without thought to the production device, with up to eight voices on hardware supporting that number. Table 1 shows a memory map for the routines running under DOS 3.3.

At this point, the list of supported players includes: Apple Speaker (1 voice), Temporal Acuity DAC (4 voice), Software Automatic Mouth (4 voice), Mockingboard/A.E. Phasor, Passport MIDI interface, and Roland MIDI interface (not on the IIGS). The routines are located upwards from $6000, so you can also develop software for (and on) the Apple II+. I have even managed to use the SAM speech routines in the upper 16K of a 64K machine, in a program having 'Spoken Prompts' along with music through the SAM and TAP DAC boards! In short, this is a multi-hardware supporting concept.

In Toolbox format, most music code consists of 3 bytes of information per note, representing Rhythm, Pitch, and Display characteristics. The documentation contains tables of basic values, to which you add modifying values. Actually, you are setting additional bits for features such as stem direction, showing accidentals, etc. A sample program is provided on the disk that uses data statements to hold the values for your score, which are then converted into a binary file for use by the Toolbox. It is also possible to use an assembler to generate the files, as long as you generate the table of lookup pointers through DA opcodes.

The software routines use a parameter page containing default settings, which can also be over-ridden through passed parameters. The Basic interface uses the Ampersand hook (or direct calls) to pass parameters to the play and display routines. Listing 1 and Figure 4 shows a BASIC program, and a dump of the screen display which results. Since this is an open system, there is also a provision for *user added* routines which are parsed after the initial sequence.

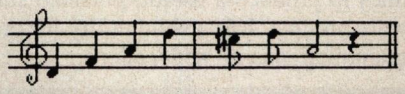
The hi-res text handling routines possess a remarkable similarity to BASIC Text Screen commands. VTAB and HTAB values are read and used, and you have an INVERSE command with a nice twist — the character height is set at 10 pixels, so the entire character is surrounded by inverse. The format of the character tables is clearly described, too, so that you can add your own custom shapes as needed.

The MIDI interface possesses not only playback capability, but the ability to 'Read Notes' from the MIDI keyboard, *while* playing music out. The possibilities this raises for interactive music software could open a whole new approach to educational programming. Actually, each of the hardware manufacturers has equivalent routines that are

TABLE 1
Toolbox Memory Map

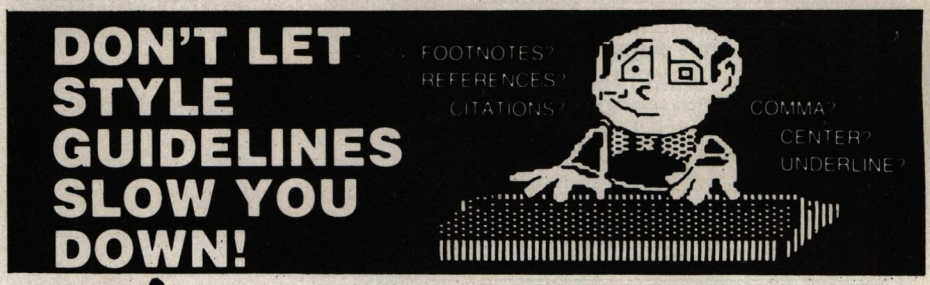
Address	Use
$9600	DOS
$9000	Music Score Tables
$8D00	Text Tables
$8100	Graphics Characters
$8000	Waveform
$7400	Sound Drivers
$7300	Sound Variable Page
$6100	Toolbox Kernel Routines
$6000	Toolbox Variable Page
$4000	Hi-Res Page 2
$800	Applesoft Program Space
$0	Apple Stuff

FIGURE 4
Musician's Toolbox Sample Screen



LISTING 1

```
10  REM Musician's Toolbox Demo
20  REM
30  PRINT CHR$ (4);"BLOAD SAMPLE.SCORES'
40  & CLEAR
41  REM CLEAR & DISPLAY HGR2 SCREEN
42  REM
50  & PLOT 0,L = 5,R = 34,T = 50
51  REM DRAW STAFF (TREBLE CLEF)
52  REM LEFT + RIGHT MARGINS, TOP LINE
53  REM
60  & DRAW 0,V = 1,L = 8
61  REM DRAW MELODY 0, ONE VOICE
62  REM BEGINNING AT COLUMN 8
63  REM
70  VTAB 20: HTAB 13: & PRINT "<P> TO PLAY"
71  REM  PRINT TEXT ON SCREEN
72  REM
80  GET A$: IF A$ < > "P" AND A$ < > "p" THEN 80
81  REM
90  & LOAD 0,V = 1,TM = 80
91  REM PLAY MELODY 0, 1 VOICE
```

Manuscript Manager™

APA and CBE Style Made Easy

". . . run, do not walk, to buy this one."
—Marc Apfelstadt, Ph.D., **Call-A.P.P.L.E.**

". . . a joy to work with . . ."
—Chris Alberton, **II Computing**

Manuscript Manager™ is a one-of-a-kind software package that completely automates all of the style guidelines of the APA (American Psychological Association) or the CBE (Council of Biology Editors) including References, Citations, Footnotes, Abstracts, Tables, and Equations.

This complete word processor, designed for the Apple IIe/c, lets you create, edit and print a document that accurately follows these guidelines.

Order Now, for the Apple IIe and IIc:
Manuscript Manager: APA Style $175.00
Manuscript Manager: CBE Style $195.00
Demonstration Disks are available at $10.00 each

Pergamon Software A Division of Pergamon Journals, Inc.
A member of the BPCC PLC Group of companies
Maxwell House, Fairview Park, Elmsford, New York 10523 (914) 592-7700

CIRCLE 11 ON READER SERVICE CARD

machine specific, but this is *portable*.

There are just two minor 'features' with the software. On a color monitor, the 'color demon' surfaces in the form of mixed green and purple shadings on the music display at times. The characters are clearly drawn, so it is really a minor inconvenience when weighed against the programming flexibility.

At this point, there is not a sound driver available for the IIGS Ensoniq chip. Actually, with the open nature of the system and the clear definition of the pointers, it would probably be an interesting project for an A.P.P.L.E. Co-op member to adapt a public domain GS driver to play from Toolbox format & command structure. The authors opted to get the tools out in the present state, rather than causing any further delay. Since two are pursuing doctoral studies in music (and one is in pursuit of tenure!), you can certainly understand the presence of other commitments. *If you get a working GS driver running under the system, please be sure to let me know, and I can contact them for you.*

The authors have some unusual stipulations for use of the software, based on their interest in music programming. If you desire to publish software using the play & display routines, there is *no fee* involved, simply three major conditions to meet. The first, screen and documentation credits, almost goes without saying. The second is that the author and publisher each own a purchased set of the software... no problem at this price! The third stipulation is that 4 sets of the software be provided through the publisher (Wm. C. Brown), which will be distributed to the authors — they're interested in seeing the products that grow out of this project! The simple registration procedure is clearly spelled out in an appendix to the documentation, and the policy is a refreshing change. Hats off for that decision!

I know this set of utilities very well, since I used an early version of the software to write my own nine disk package of ear-training software [*Music Fundamentals*, Wm. C. Brown, 1986]. From first hand experience, I can say that anyone with a modest knowledge of BASIC programming and a grasp of musical concepts can apply these routines to produce a useful, polished software product. It is a solid utility for programming musical applications.

In Summary:

Each of these programs possesses a unique strength as a music application. Deluxe Music Construction Set offers IIGS sound combined with above average printout of piano type musical scores, while The Music Studio provides tremendously powerful and flexible sound output. The SuperSonic Stereo card and Digitizer are true enhancements to the basic IIGS hardware, and Future Sound has a lot of potential for digital recording. Music Printer gives phenomenal printed output, and The Musician's Toolbox places flexible programming tools at your disposal at a super price. I can honestly recommend each of these products for their strengths... they are true enhancements to the use of a *musical II*!

Please rate this article on the Reader Service Card by circling:

41 Excellent **42** Good

43 Fair **44** Poor

For More Information

Product:
Music Construction Set for IIGS
Company:
Electronic Arts
l820 Gateway Drive
San Mateo, CA 94404
(415) 572-2787
Price: $49.95
CIRCLE **35** ON READER SERVICE CARD
(Also available through A.P.P.L.E. Co-op, see page 5.)

Product:
The Music Studio (GS)
Company:
Activision, Inc.
P.O. Box 7287
Mountain View, CA 94039
(415) 960-0410
Price: $79.95
CIRCLE **36** ON READER SERVICE CARD

Product:
Future Sound (GS)
Company:
Applied Visions, Inc.
Suit 2200
One Kendall Square
Cambridge, MA 02139
(617) 494-5417
Price: $175.00
CIRCLE **37** ON READER SERVICE CARD

Products:
Supersonic Stereo Card (GS)
Supersonic Digitizer (GS)
Company:
MD Ideas
1163 Triton Drive
Foster City, CA 94404
(415) 573-0580
Price:
Stereo Card $59.95
Digitizer $39.95
CIRCLE **38** ON READER SERVICE CARD

Product:
Music Printer (II - IIGS)
Company:
Temporal Acuity Products
Building 1, Suite 200
300-120th Ave. NE
Bellevue, WA 98005
(206) 462-1007
Price: $149.95
CIRCLE **39** ON READER SERVICE CARD

Product:
The Musician's Toolbox (II - IIGS)
Company:
Wm. C. Brown Publishers
2460 Kerper Boulevard
Dubuque, IA 52004-0539
(319) 589-2814
Price: $39.95
CIRCLE **40** ON READER SERVICE CARD

Precision Data Products™
POLY PACK 5¼" DISKETTES
(From Leading Mfr.)

36¢ Each

- Blank Jackets
- WP Tabs
- Envelopes

100% Error Free
Lifetime Warranty
Sold in Lots of 100 Only

SALE!
SONY POLY PACK 3.5" DISKETTES

High Capacity Data Storage. Meets the requirements of all 3.5" Micro Diskette Drives.

SS 135TPI	$1.17 ea.
DS 135TPI	$1.29 ea.

PRINTER RIBBONS
Quality replacements for most popular printers.

Apple Imagewriter	Black	$3.95 ea.
Apple Scribe	Black	$3.95 ea.
Epson LX 80/90	Black	$2.95 ea.
Okidata 80/82/83	Black	$1.75 ea.
Toshiba 1350	Black	$4.89 ea.

Color Ribbons Available At Great Savings Too

Min. Order $25.00. Add 10% for less than 50 disks. S&H: Continental USA $4.00/100 or fewer disks. $2.00 per dozen ribbons. Reduced shipping charge on larger quantities. Foreign orders, APO/FPO, please call. MI residents add 4% tax. Prices subject to change without notice. Hours: 8:30 AM - 7:00 PM ET.

CALL TOLL FREE
1-800-258-0028

Precision Data Products™
P.O. Box 8367
Grand Rapids, MI 49518
(616) 452-3457 • MI 1-800-632-2468

CIRCLE **12** ON READER SERVICE CARD

Peeking at and Poking Around the Apple IIGS Monitor

This is a two part article that takes a look at the Apple IIGS monitor. It is based on material in Michael Fischer's *Apple IIGS Technical Reference*. The article contains a road map of what is in the monitor as well as the rules of the road to use in your explorations. It also discusses the differences between the Apple IIGS monitor and earlier members of the Apple II family.

Part 1 introduces the monitor, discusses entering values, and examining and changing memory and registers. It also describes the use of the disassembler.

Part 2 tells you how to use the monitor to run a routine, use a toolbox function, convert numbers from one base to another, perform hexadecimal arithmetic, redirect input and output, change the video-display, and use the mini-assembler and diagnostic routines.

Michael Fischer

One of the dictionary definitions of the verb "monitor" is "to keep track of, regulate or control." The IIGS system monitor is a set of subroutines located in ROM which aid you in keeping track of, regulating or controlling the IIGS.

The system monitor in the IIGS is derived from the original Apple II system monitor, written by Steve Wozniak and Allen Baum. Apple has modified this monitor over the years to add new commands and to support features added by the Apple II+, IIe and IIc. Over the same years, several features have also been dropped.

As can be expected, the IIGS system monitor is significantly changed from the existing Apple II system monitors, providing support for the additional registers, new operating modes and increased memory addressing capabilities of the 65816 mircoprocessor.

Low Level Routines

The system monitor consists of several parts. Some routines perform low level input/output operations such as reading input from the keyboard or game port, and sending output to the screen or speaker. In the earlier Apple II system monitors, these functions were a significant part of the monitor and programs would often call the monitor subroutines to perform their input/output functions.

The IIGS' ROM is four to eight times the size of the ROM in previous Apple II's. This greatly expanded ROM contains highly sophisticated routines (the "Toolbox"), not merely the low level routines found in the Apple II system monitor.

Yet, the low-level routines of the traditional Apple II system monitor exist, for the most part, in the IIGS. They are required for compatibility with existing Apple II software. A full list of the supported routines is given in Table 1. This table shows the name of the routine and its location in the ROM. These routines only include the entry points Apple has committed to support.

The Miscellaneous Toolset contains one function which aids in the use of these monitor routines from IIGS 16-bit mode. FWentry, which is function $24, takes as input the accumulator, X register and Y register values passed to the routine (as the low bytes in three different words), followed by the 16 bit address of the entry point as shown in Table 1.

TABLE 1
List of System Monitor Routines

Add	Name	Description
F800	PLOT	Plots on the low resolution graphics screen
F80E	PLOT	Modify low resolution byte on the screen
F819	HLINE	Draws low resolution horizontal line
F828	VLINE	Draws low resolution vertical line
F832	CLRSCR	Clear the low resolution screen
F836	CLRTOP	Clear the low resolution screen in mixed mode
F847	GBASCAL	Calculate base address for low resolution screen
F85F	NXTCOL	Increment low resolution color by three
F864	SETCOL	Set the low resolution color
F871	SCRN	Determine low resolution screen color
F88C	INSDS1.2	LDA (PCL,X) and goto INSDS2
F88E	INSDS2	Set up indexes for opcode in accumulator
F890	GET816LEN	Return 65816 instruction length in $2F
F8D0	INSTDSP	Display disassembled instruction
F8D9	SPATFNS	Applesoft call for HLIN
F93A	GETAN3STATUS	Transfer AN3 status to accumulator bit 7
F940	PRNTYX	Print values of X and Y registers in hex
F941	PRNTAX	Print values of accumulator and X register in hex
F944	PRNTX	Print three spaces
F94A	PRBL2	Print X register number of spaces
F953	PCADJ	Adjust program counter
F962	TEXT2COPY	Enable/disable text page 2 copy
FA40	IRQ	Interrupt request handler
FA4C	BREAK	Break handler
FA59	OLDBRK	Prints 65816 BRK printout
FA62	RESET	Reset handler
FA68	FBBYTE	$FB requires here for some code
FAA6	PWRUP	System coldstart
FABA	SLOOP	Disk controller slot search loop
FABF		Matches $00 at FAFF
FAD7	REGDSP	Display contents of register
FAFF		$00 must appear here
FB05	APPLEMSG	'Apple II' must appear here
FB19	RTBL	Register names table
FB1E	PREAD	Read a hand controller (paddle)
F821	PREAD4	Timeout paddle and call PREAD
F82F	INIT	Initialize system
FB39	SETTXT	Place system in Text mode
FB40	SETGR	Place system in Graphics mode
FB4B	SETWND	Set text window
FB51	SETWND2	Set window size
FB5B	TABV	Vertical tab
FB60	APPLEII	Print machine type
FB6F	SETPWRC	Create a power-up byte
FB78	VIDWAIT	Wait to Control-S
FB83	KBDWAIT	Wait for key press
FBB3	VERSION	Monitor ROM identification byte
FBBF	ZIDBYTE2	Monitor ROM sub identification byte #2
FBC0	ZIDBYTE	Monitor ROM sub identification byte #1
FBC1	BASCALC	Text base address calculator
FBDD	BELL1	Generate user selected tone
FBE2	BELL1.2	Branch to BELL1
FBE4	BELL2	Branch to BELL1
FBF0	STORADV	Place a printable character on screen
FBF4	ADVANCE	increment the cursor position
FBFD	VIDOUT	Place a character on the screen
FC10	BS	Back space
FC1A	UP	Move cursor up a line
FC22	VTAB	Vertical tab
FC24	VTAB2	Vertical tab (alternate entry)
FC42	CLREOF	Clear to end of page
FC58	HOME	Home cursor and clear screen
FC62	CR	Carriage return — begin a new line
FC66	LF	Line feed
FC70	SCROLL	Scroll the screen
FC9C	CLREOL	Clear to end of line
FC9E	CLEOLZ	Clear to end of line for BASL position
FCAB	WAIT	Delay
FCB4	NXTA4	Increment pointer A4
FCBA	NXTA1	Increment pointer A1
FCC9	HEADR	Write header to cassette tape
		Cassette port not available
FD0C	RDKEY	Inverse flash cursor and get input character
FD10		BASIC.SYSTEM uses for input
FD18	RDKEY1	Get an input character
FD1B	KEYIN	Read the keyboard input
FD35	RDCHAR	Get an input character of ESCape code
FD67	GETLNZ	Get an input line
FD6A	GETLN	Get an input line after prompt
FD6C	GETLN0	Alternate entry for GETLN
FD6F	GETLN1	Get an input line without prompt
FD8B	CROUT1	Print a carriage return and clear to end of line
FD8E	CROUT	Print a carriage return
FD92	PRA1	Print carriage return and A1 in hex
FDDA	PRBTYE	Print value of accumulator in hex
FDE3	PRHEX	Print low nybble of accumulator in hex
FDED	COUT	Output a character through the output' hooks
FDF0	COUT1	Output a character to the screen
FDF6	COUTZ	Output a character to the screen
FE1F	NEWID	New system identification routine
FE20	MOVE	Move a block of memory
FE80	SETINV	Set inverse text mode
FE84	SETNORM	Set normal text mode
FE89	SETKBD	Reset input to keyboard
FE8B	INPORT	Reset input to slot
FE93	SETVID	Reset output to screen
FE95	OUTPORT	Reset output to slot
FEB6	GO	Begin code execution
FECD	WRITE	Write a record to cassette
FEFD	READ	Read a record from cassette
FF2D	PRERR	Print ERR and sound beep
FF3A	BELL	Call BELL1 to generate beep
FF3F	RESTORE	Restore all registers from stored locations
FF4A	SAVE	Save all registers in set locations
FF58	IORTS	Known RTS ($60) for 'use
FF59	OLDRST	Old entry point to enter monitor
FF65	MON	Standard monitor entry point with beep
FF69	MONZ	Standard monitor entry point
FF6C	MONZ2	Monitor entry point - registers intact
FF70	MONZ4	Act on command in input buffer ($200-2FF)
FF8A	DIG	Shift hexadecimal digit into A2
FFA7	GETNUM	Transfer hexadecimal input into A2
FFAD	NXTCHR	Translate next character
FFBE	TOSUB	Execute command pointer to by Y register
FFC7	ZMODE	Clear mode and set Y register to 0
FFCB	PRODOSRTS	RTS ($60) used by ProDOS
FFD0	DOSEF	DOS use
FFF6	CHKSUMTWEAK+1	Tweaked to make checksum= $1234

TABLE 2
Monitor Memory Examination Commands

BAddress.EAddress	Displays memory from BAddress to EAddress
<Return>	Displays next 8 (in 40 column mode) or 16 (in 80 column mode) memory locations
Add1S<Add2S.Add2EV	Verify if block of memory from Add2S to Add2E is the same as that block beginning at Add1S.
\Pattern\<RangesS.RangeEP	Search for pattern in range beginning at RangeS and ending at RangeE.
L	List (disassemble) assembly language code.
Control-E	Examine 65816 and IIgs registers
=T	Display system time and date.

FIGURE 1
Memory Display

```
*FF/3C90:04 D0 85 C2 30 A2 84 00-.P5B0"
```

Address — 1-byte hexadecimal values — ASCII code values

Values returned by the function include the Processor Status, Accumulator, X and Y registers as the low bytes in four different words.

The monitor also contains video support routines, methods by which you can adjust the video display of the computer. These routines handle the display when in text, low resolution graphics and high resolution graphics mode. The Quickdraw][, and related tools, handle the video output display when operating in super resolution graphics. The monitor video support routines are basically unchanged from the Apple IIe and IIc.

Commands

The system monitor also contains a program that is run whenever you enter the monitor through its main entry point ($FF/FF69). This area of memory is normally shadowed onto bank 0 when the IIGS is in 8-bit Apple II mode. These routines provide a method of determining and changing the values in various places in the IIGS memory or the 65816 processor, running machine language programs and performing mathematical and miscellaneous other functions.

This program is greatly expanded from that on the Apple IIe and IIc. The commands fall into six different categories. Some of these commands also exist in the Apple Programmer's Workshop debugger (available from APDA). The debugger commands, when available, are more sophisticated and more integrated than the monitor routines.

Before giving a command to the system monitor, you must first place the IIGS in the system monitor level operating mode. The mode is indicated by the use of the "*" for a prompt character. To enter the system monitor, you must call the entry routine, which is located at $FF/FF69. From Applesoft BASIC (run BASIC.SYSTEM from the Launcher) this can be accomplished by typing "CALL -151".

Value Specification

Address locations in the IIGS monitor are given in hexadecimal (base 16) notation. The address consists of two parts, a one byte bank number separated with a slash (/) from a two byte bank address. In most cases, if the bank number is not used, the last accessed bank is assumed.

You can specify values in the monitor by entering:

1. Addresses and bank numbers as hexadecimal numbers with or without the $ prefix which normally signifies a hex value. If fewer than two digits for a one byte value or four digits for a two byte value are entered, the value is padded on the left with one or more 0's (e.g., the entry "3E2" is interpreted as "03E2".)
2. Hexadecimal values of up to eight bytes in length (except for addresses and bank numbers). You can enter multiple values on the same line if you separate them with a space. If the total number of digits entered is a odd number, a 0 is added on the left end. The values of bytes entered without spaces are flipped when stored so that the least significant byte is placed is memory first. Thus the value 34F41E is placed into memory with 1E in the lowest address, F4 in the next address and 34 in the highest address.
3. An ASCII character in place of a hexadecimal number. Two forms of ASCII input are supported — literal and flipped. In both cases the negative ASCII value (i.e., high bit set) of the character is placed into memory after it is ANDed with the ASCII filter mask.

Literal ASCII, which is delimited by quote marks ("), is placed into memory in the order entered, i.e., the value of the leftmost character is placed in the lowest memory address. The maximum length of a literal ASCII entry is 255 characters.

Flipped ASCII, which is delimited by single quote marks ('), is placed into memory in the reverse order of entry, i.e., the value of the rightmost character is placed in the lowest memory address. The maximum length of a flipped ASCII entry is 4 characters — if you enter more than four characters, the monitor uses the last four entered.

The Apple IIe and Apple IIc monitors do not support flipped HEX or ASCII input. Literal ASCII input is restricted to single character values and the character has to be preceded by a single quote ('). There is no AND mask and the high bit of the ASCII character is always set.

Examining the System

There are seven different commands which examine the status of either memory or a 65816 register on the IIGS. (See Table 2 for a listing.) Memory examination commands make use of the concept of the last opened location, — the most recently examined memory location. If you press the return key at the monitor prompt, the monitor displays the contents of up to 16 consecutive memory locations starting just after the last opened location.

The exact number of locations displayed by pressing the return key depends upon whether the IIGS is in 40 or 80 column text mode and what the last opened location was. In 40 column mode, the IIGS displays up to eight values, ending with an address whose last digit is either $7 or $F. In 80 column mode, the IIGS displays up to 16 values, ending with an address whose last digit is $F. At the end of a bank ($xx/FFFF), the display wraps around to the beginning of the same bank ($xx/0000) and does not span the bank boundary.

The memory display is illustrated in Figure 1 and consists of three parts. The beginning address location, which is one number higher than the last opened location, has a one byte

bank number, a "/" as a separator character, a two byte address, and a colon (:). The one byte hexadecimal value for each memory location, separated by spaces, follow the address. Next comes a dash followed by the one character ASCII representation of each location. Values which do not have a printable ASCII character representation display a period.

To display memory at a specific location other than the last opened location, you merely enter the address and press the return key. To display a specific range of addresses, enter the beginning address, a period (.), and the ending address. For an address, you use either a one byte bank number, a slash (/), and a 2 byte value, or just a 2 byte value in which case the current bank number is assumed. Do not enter the hex prefix symbol ($). A range must be within one bank. If the ending range value has a bank number different from the beginning bank number, the ending bank number is ignored.

If you have entered a display range with a large number of values, and wish to interrupt the display, you can enter a control-X to stop the display. A control-S will temporarily freeze the display until another key is pressed.

The behavior of the memory display function on the IIGS monitor parallels that of the Apple IIe and IIc except for the addition of the bank address value and the display of 16 values in 80 column mode. The ASCII display was added in the Apple IIc and enhanced Apple IIe monitors.

Verify (V) Command

The V (for verify) command determines if the contents of two different memory ranges are the same. Its form is:

```
{Address 1 Start}<{Address 2 Start}.{Address 2 End}V.
```

This command compares the values in the range specified by Address 2 Start and Address 2 End with the correlative values in the range beginning with Address 1 Start.

If the two ranges have the same values, the monitor prompt (*) value returns without any other output. If there are differences between the two ranges, the locations with different values are displayed as {Address 1 Location}-{Address 1 Value} ({Address 2 Value}). The V command displays all non-matching locations.

Assume that the following command is given:

```
*10/3456<11/2356.11/2456V
```

and the values in the two ranges are identical except that 10/3458 contains a 4F while 11/2358 contains a 3C. The result of the command would be:

```
*10/3458-4F (3C)
```

The V command is identical to the V command on the Apple IIe and IIc except for the bank part of the address.

Pattern (P) Command

A close relative to the verify command is the P (for Pattern) command which determines if a specified pattern is present in a given range of memory. The pattern can be up to 236 bytes long and consist of either hexadecimal numbers, literal ASCII, or flipped ASCII. If the pattern is not found, the computer returns you to the monitor prompt (*) without any message. If the pattern is found, each starting address for the pattern is displayed on the screen. The format for the command is:

```
\{pattern}\<{range start}.{range.end}P
```

The search range is restricted to a single bank.

Assume you want to determine if the pattern 49 49 47 53 (which is the ASCII code for "IIGS") is in bank $01 of your IIGS. To do this, you would enter the command:

```
\49 49 47 53\01/0000.FFFFP
```

or you could enter the equivalent command:

```
\"IIGS"\01/0000.FFFFP
```

If the pattern exists starting at 01/279C and at 01/342D, the monitor response to this command is:

```
*01/279C
*01/342D
*
```

On the enhanced Apple IIe and 32K ROMs in the Apple IIc, the pattern command is known as the S (for search) command. It is restricted to a two byte hexadecimal value which is always flipped. The command is not available on earlier versions of the Apple IIe and IIc.

Disassembly (L) Command

The contents of memory on the IIGS, as on most computers, can also be interpreted as a part of a machine language program. The L (for list) command interprets memory in this way. This process of taking machine language code and converting it to an assembly language source listing is called disassembly.

The L command in the IIGS system monitor takes the contents of memory beginning at the location you specify and processes enough bytes to produce sufficient assembly language lines to fill the current window. Each line of disassembly is up to 40 characters across and the disassembly display consists of a header line showing the machine state, a blank separator line, up to 20 lines of disassembly, another blank separator, and the monitor prompt character at the bottom of the screen (Figure 2). If the screen window is less than 24 lines, the number of lines of disassembly is decreased accordingly.

Each line of the main disassembly contains three parts (Figure 3). The address consists of a bank number separated from the address by a slash "/". A colon and a space follow the address. Next comes the machine language code which is from one to four hexadecimal bytes with a space between each byte. The disassembled source code, starting in column 16, includes the operation code and, where applicable, the operand. The opcodes and operands follow standard 65816 format with the following exceptions:

1. No $ prefix is used to denote hexadecimal values. All values are presumed to be hexadecimal values.
2. The amount of the offset of a branch instruction follows that instruction (in parentheses).
3. The ProDOS/8 machine language interface call, which is a JSR $BF00 command when in bank zero, is displayed on three lines as follows, only when the instruction is found in bank zero.

```
JSR     BF00      ProDOS/8 MLI call
        00        Command number
        1234      Pointer to parameter table
```

4. The ProDOS/16 machine language interface call, which is a JSL $E1/00A8 from any bank, is displayed on three lines as follows:

```
JSL     E100A8    ProDOS/16 MLI call
        0456      Command word
        005FD321  Double word pointer m to
                  parameter table
```

The L command differs from the L command in the Apple IIe and IIc monitor in several respects:

1. The bank number is part of the address.
2. A colon rather than a dash follows the address.
3. Up to ten bytes per command are used if the ProDOS/16 MLI call is involved.
4. The full 65816 command set is supported.
5. The offset follows branch locations.
6. There is no $ hexadecimal prefix.
7. Shorter vertical text windows are automatically accommodated.

FIGURE 2
Disassembly Screen Display

```
Machine state          0=m 0=x 1=LC bank (0/1)
Separato
Disassembly listing    FF/3C90:04D0  TSB D0
                       (19 more lines go here)
Separator
Monitor prompt         *
```

FIGURE 3
Disassembled Line

FF/3C94:	30 A2	BMI 3C38	(-5E)
Address	Machine language code	Disassembled code	Branch offset (if applicable)

TABLE 3
Register Labels Used by Control-E Command

A	Accumulator
B	Data bank register
D	Direct register
e	Emulation bit
K	Program bank register
L	Language card bank
M	Machine estate
m	Memory width bit
P	Processor status register
Q	Quagmire state
S	Stack pointer
X	X register
x	Index register width bit
Y	Y register

FIGURE 4
Display of Registers In 80-Column Mode

```
A=BF6F X=86FF Y=0000 S=86FB D=0000 P=81 B=00 K=00 M=0C Q=80 L=1 m=0 x=0 e=0
```

8. The ProDOS/8 and ProDOS/16 MLI calls are specially displayed.

Examine Register Values

It is important to be able to check the values in the various 65816 registers as well as the values in the memory locations of the IIGS. The examine registers command, Control-E, displays the contents of all the 65816 registers as well as several IIGS softswitch settings as well. The display occupies one screen line in 80 column mode and two lines in 40 column mode. Figure 4 shows the register display in 80 column mode in native mode and with 16 bit accumulator and index registers.

The meaning of the register labels are given in Table 3. Note that the program counter is not displayed by the examine command. The examine command in the Apple IIe and IIc monitor shows only the registers available on the 6502/65C02, namely A, X, Y, P and S. Each register is only 8 bits wide. The command also permits changing the values in the registers. Later, there will be a discussion of how to change the 65816 registers using the IIGS monitor.

Time Display

The IIGS has a built-in real time clock. The display time (=T) command displays the current system date and time as follows:

```
Time= MM/DD/YY HH:MM:SS AM
```

Since the Apple IIe and IIc do not have a built-in real time clock, there is no Apple IIe or IIc monitor command relating to time.

Modifying Memory

The second major category of system monitor commands is the modification commands which allow you to change the values in memory locations or 65816 registers. Like the main memory examination command discussed earlier, the main memory modification command uses the concept of the last opened location, also known as the next changeable location. Entering a colon followed by values, will insert those values in the IIGS's memory beginning which the next changeable location. The values may be either hexadecimal, flipped hexadecimal, literal ASCII, or flipped ASCII. If multiple values are to be entered, they should be separated by spaces. Multiple flipped hexadecimal, or ASCII, can be entered if the various groups of values are separated by spaces.

If you wish to enter values at a location other than the nextchangeable location, or if you do not know what the next changeable location is, you can specify a starting location for the insertion of values. To do this, you enter an address in the standard IIGS monitor format, then a colon, and then the values to be entered.

Move Command

Another way to change memory values is by the M (for move) command. Technically speaking, this command does not perform a move (delete data from source location), but a copy (leave data in source location). The distinction between move and copy is quite fuzzy in the 65816 world because the

65816 MVN and MVP commands, which copy blocks of memory, are called move commands by the 65816 literature.

The format of the move command is:

```
{destination.start}<{source.start}.{source.end}M
```

When given this command, the monitor takes the byte at source.start and copies it to destination.start. It then takes the next higher byte from the source block and copies it to the next location in the destination block, continuing until it reaches source.end.

A useful method of remembering where the source and where the destination address goes in the command is to think of the < as an arrowhead pointing to the direction of movement of the values.

The move command can be used to place a repeating pattern in memory. To do this, place the pattern in memory, using the : command, beginning with the first location in which the pattern is to appear. Then issue the move command using the location in which the next repetition of the pattern is to appear as the destination.start. The place where you put the start of the pattern is the source.start address. To compute the source.end location multiply the number of bytes in the pattern by the number of times you wish the pattern to repeat less one, subtract one, and add the result to the source.start location.

By way of illustration, to place the 4 byte pattern 02 ED 4F 37 into memory beginning at location 12/3459 for a total of 8 times you would enter the following:

```
*12/3459:374FED02
```

[Place bytes in memory. Multi-byte hexadecimal input is flipped.]

Compute values as follows:

```
Destination.start    =12/345D (3459 + 4)
Source.start         =12/3459
Source.end           =12/3474 (7 repetitions (8-1)
                       times 4 bytes = $1C bytes
                       $1C - 1 = 1B.
                       3459 + 1B = 3474.)
```

Then enter:

```
*12/345D<3459.3474M
```

Zap Command

Where the repeating pattern consists of a single byte value, there is another method of placing that byte in memory. The Z (for Zap) command places a one byte value throughout a specified memory block. The format of the command is:

```
{val}<{block.start}.{block.end}Z
```

Change Register Values

The value in any of the 65816 registers may be modified by using the appropriate monitor command. Other items which can be changed through similar commands include some of the IIGS softswitches, memory shadowing, machine speed, language card bank, ASCII filter mask, and three status registers bits — m (accumulator/memory width), x (index register width) and e (emulation or native mode switch). The basic format of these commands is:

```
{val}={register.symbol}
```

Table 4 gives a listing of the various register symbols and for each indicates whether the value should be one or two bytes.

As mentioned above, the setting of certain machine state indicators, in addition to the 65816 registers, can be modified. The machine state is a one byte value which indicates the state of eight different IIGS softswitches. The purpose of each bit is shown in Table 5. The machine state value is modified by:

```
{val}=M
```

The softswitch which controls which of the two 4K "language card" or D000 RAM banks are active is controlled by the L or language card bank value. A 0 indicates the first bank (sometimes called bank zero and sometimes called bank one) is active; a 1 indicates the second bank (sometimes called bank one and sometimes called bank two) is active. (Note: when used in this context, "bank" refers to a 4 kilobyte range of memory mapped to $E0/D000 or E1/D000 through $DFFF. It does not mean a 64 kilobyte range of memory, which is the common usage of bank in 65816 terminology.) The language card state is set by:

```
{value}=L
```

Value should be either 0 or 1. Otherwise the least significant bit is taken as the value.

The quagmire state, which indicates the status of memory shadowing, can be changed by the following command:

```
{val}=Q
```

The purpose of the one-byte ASCII filter mask is discussed earlier in this article under ASCII input. Its value is changed by:

```
{val}=F
```

The status of three processor status register bits can also be individually modified. (The =P command modifies all eight status bits.) The value passed to the command must be either 0 or 1 or it is ignored. The general format of the command is:

```
{value}={bit symbol}
```

The symbol used for each bit, and the meaning of that symbol, are given in Table 6.

The various registers, machine state and similar values discussed above are not immediately changed when the change command is given. The program itself requires certain values in the registers and certain machine states. A change in these values could crash the monitor. Instead, the change commands modify the values which are stored in specified RAM memory locations. When one of the monitor run commands is given, these values are placed in the 65816 registers and appropriate changes made to machine states before control is passed to the called program. These stored values, especially the m and x bits, language card value, quagmire state and machine state are also used by the disassembly and display memory commands.

TABLE 4
Labels Used to Set Various Registers

A -	Accumulator (1 or 2 bytes)
B -	Data bank register (1 byte)
D -	Direct register (2 bytes)
e -	Emulation bit (0 or 1)
K -	Program bank register (1 byte)
L -	Language card bank (0 or 1)
M -	Machine state (1 byte)
m -	Memory width bit (0 or 1)
P -	Processor status register (1 byte)
Q -	Quagmire state (1 byte)
S -	Stack pointer (1 or 2 bytes)
X -	X register (1 or 2 bytes)
x -	Index register width bit (0 or 1)
Y -	Y register (1 or 2 bytes)

TABLE 5
Meaning of Bits in Machine State Byte

Bit#	Meaning
0	INTXCROM active if set
1	Alternate ROMBANK active if set
2	Language Card Bank 2 active if set
3	RDLCROM active (for read only) if set
4	RAMWRT active if set
5	RAMRD active if set
6	Page 2 active if set
7	Alternative zero page active if set

TABLE 6
Meaning of bit state symbols

e -	Emulation bit in 65816 Processor
m -	Memory width bit in 65816 Processor
x -	Index register width bit in 65816 Processor

TABLE 7
Standard Configuration Given by Control-R Command

A -	0000
B -	00
D -	0000
e -	1
K -	00
L -	1
M -	0C
m -	1
P -	00
Q -	00
S -	01F2
X -	0000
x -	1
Y -	0000

TABLE 8
Range for Time Command Values

Hour -	00-23
Minute -	00-59
Second -	00-59
Month -	1-12
Day -	1-31
Year -	00-99

Restore Command

The various values stored for register and status change can be reset to the standard monitor configuration by the use of the Control-R (for Restore) command. This sets the values to those in existence upon entrance to the monitor. These values are listed in Table 7.

The setting of the IIGS's real time clock can be changed either by use of the control panel or by use of the time setting command from the monitor. The format of the command is:

```
=T=mon/day/year hour:minute:second
```

Any delimiter except a single quote may be used between the values entered. The range for the various time values is given in Table 8.

The standard cursor shape in the IIGS monitor is a checkerboard in 40 column mode without the 80 column firmware active and a solid rectangle in either 40 or 80 column mode with the 80 column firmware active. This character can be changed to one you specify by entering a Control-^ (pressing and holding control and shift and then pressing the "6/^" key) followed by the character you wish the cursor to be. To change back to the standard cursor enter Control-^ followed by the delete key.

Please rate this article on the Reader Service Card by circling:

51 Excellent **52** Good **53** Fair **54** Poor

infoCache

Now ... an easy to use database for the **Apple II** with the power ... speed ... features ... to meet *your* needs:

- *custom reports*
- *personalized letters*
- *mailing labels*
- *specialized forms*
- *file management*
- *ultra fast sorting*
- *single format data entry*
- *automatic index*
- *file merge*
- *import - export utilities*

Manage ... your data faster, with less effort than you thought possible!

Design ... your files in any way *you* wish, to meet *your* needs!

see your dealer or call the

.MENU®
THE INTERNATIONAL DATABASE
1-800-THE-MENU

handcrafted software from **macroSolutions, inc.**
P. O. Box 2136
Boise, Idaho 83702

CIRCLE **8** ON READER SERVICE CARD

A Simple Periodic Interrupt Card for the Apple II

Peter & Allen Baum

While developing the desk accessory program Jeeves for the Apple II, we found that we needed a method to periodically 'wake up' our program in a way that wouldn't interfere with the main program. Our program is a background application that runs independently of the main application program. The method we chose for waking up this background task, without user intervention, utilizes interrupts.

An interrupt is used to signal the computer that some device or process needs attention. An interrupt can occur asynchronously such as... Uh, hold on for a few seconds, the phone is ringing...

Now, where were we? Oh, yes, an interrupt can occur at seemingly random times, just like phone calls, or interrupts can occur periodically, like a watch that chimes every hour. The system has some control over when it will pay attention to interrupts, just as we do when we take the phone off the hook. This process, called disabling interrupts, doesn't prevent interrupts from occurring, however it does permit the system to ignore the interrupt until the system is ready to recognize it.

An interrupt causes the processor to stop what it is doing and jump to another task. A hardware device generates an interrupt by pulling low one of two interrupt request lines on the 6502. These hardware devices usually reside in one of the regular peripheral slots, however on the IIGS and the IIc there are some internal devices, such as the serial ports, which can generate an interrupt. Although there are two interrupt request lines on the 6502, only one, the IRQ (Interrupt ReQuest) line, should be used in Apple II systems. The other signal, called NMI (Non-Maskable Interrupt), can be very destructive if it is used during critical periods, such as when reading or writing to a floppy disk.

Two Types of Interrupts

The IRQ signal can be 'masked' or disabled during critical periods by the processor. While this feature is very useful to ensure a reliable system, it does present another problem. If interrupts are disabled for too long, then the processor may lose data by not responding in time. For example, if the processor doesn't read the data coming in from either a serial port or from AppleTalk before more data comes in, then the original data will be overwritten and lost.

The NMI line is 'non-maskable' and cannot be disabled by the processor. If this line is used to interrupt the processor during a critical timing loop, such as when writing to a disk, then the disk data may be corrupted. For example, while the processor is handling the NMI interrupt, the disk continues to spin. When the NMI routine has been completed, the processor returns to the disk write operation, but the disk head is now at a different place on the disk. The write operation continues and overwrites the wrong data, destroying it.

When an IRQ interrupt occurs and processor interrupts are enabled, the processor jumps to the instruction whose address is stored in locations $FFFE-$FFFF. This address (also called a vector) is the start of the System Monitor's interrupt handler. The system interrupt handler will save the state of the machine, such as the memory bank configuration and the register contents, and then attempt to identify which device caused the interrupt. If none of the internal devices caused the interrupt, which must be the case in the IIe and the II+, then a branch is taken to the address stored in the IRQ vector at locations $3FE-$3FF. This is the address of the ProDOS interrupt handler, which will take over and attempt to identify the interrupting device.

Within ProDOS, the interrupt handlers are checked, one by one, in the order in which they were installed. If the handler doesn't recognize the interrupt, then it notifies ProDOS to continue with the next handler. When the handler of the device that caused the interrupt is called, it 'claims' the interrupt, processes it, clears it, and indicates to ProDOS that it has been serviced. ProDOS then returns to the monitor, which restores the machine back to its original state and returns to the original application (which is totally unaware that all this has occurred). If the interrupt remains unclaimed after all the handlers have been checked, then ProDOS displays an error message such as "INSERT SYSTEM DISK AND RESTART -ERR 01". As you can see from this process, the device priority is handled in software.

Interrupt Handler

Installation & Requirements

The first task of a program that uses interrupts is to identify the slot of the interrupting device. This information is used to configure the interrupt handler which is then installed in ProDOS. Next, the device's interrupt is enabled, and finally, the 6502 interrupt is enabled using the CLI instruction (CLear Interrupt disable processor status bit).

The interrupt handler must:

0) Be installed into ProDOS using the ALLOC_INTERRUPT call.
1) Reside in main memory (not auxiliary memory on the IIe).
2) Start with the CLD instruction (Someday we'll find out why).
3) Return via an RTS with Carry Set if the handler didn't process the interrupt.
4) Return via an RTS with Carry Clear if the handler processed the interrupt and CLEARED the interrupt hardware.

When the program is finished using the interrupt device, it should remove the interrupt handler. To prevent an errant interrupt from crashing the system when removing the interrupt handler, follow the steps below, in order:

1) Disable processor interrupts.
2) Disable and clear interrupts on the device.
3) Use the DEALLOC.INTERRUPT call to remove the interrupt handler from ProDOS.

Interrupting Hardware

An interface card that supports interrupts should provide the following:

0) An identification scheme that allows software to recognize the card.
1) Hardware that clears and disables interrupts when reset.
2) An open-collector driver to drive the IRQ line.
3) A method to detect whether the interface card caused the interrupt.
4) Optional support for daisy-chain protocol.

The daisy-chain protocol allows hardware devices to specify device priority in hardware, not software.

An Example

The Jeeves Interrupt Card

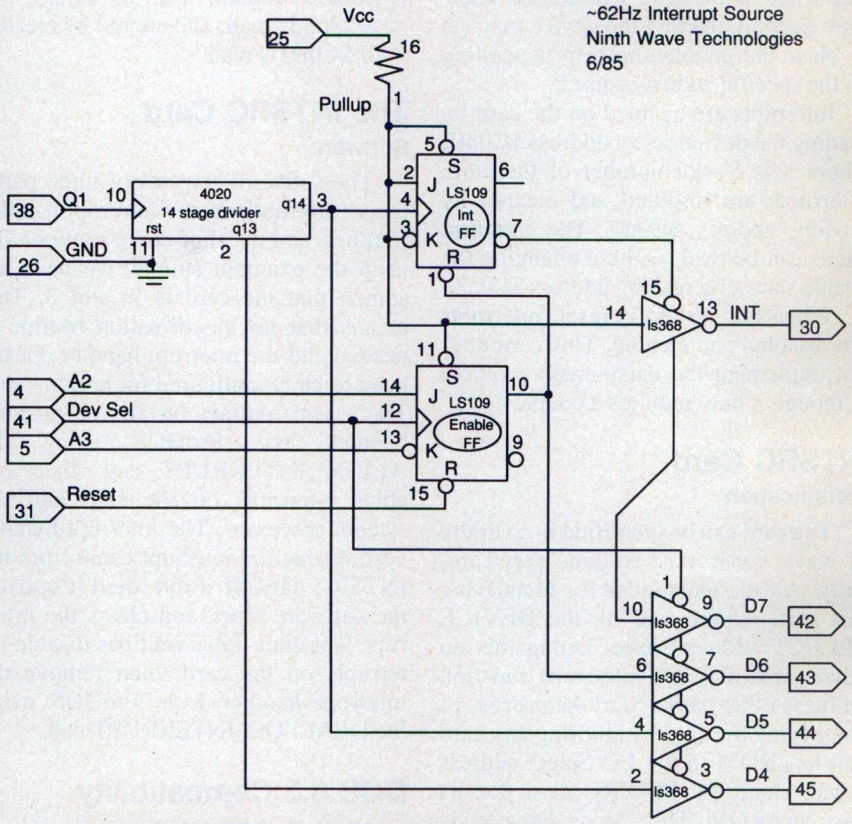
We decided that for Jeeves it would be very handy to have a periodic interrupt which we could use to implement an alarm clock and to check for special keystrokes. We designed a simple three chip card which generates an interrupt 3737 times a minute (approximately 62 times/second). This card works in any regular slot (1-7), including slot 3 of the IIe, even if the 80-column card is present in the auxiliary slot.

TABLE 1

74LS109 Function Table

	S	R	Clock	J	K	Q
Asnychronous Set	0					1
Asynchronous Clear	0					0
Load '0' (clear)	1	1	^	0	0	0
Load '1' (set)	1	1	^	1	1	1
Hold (no change)	1	1	^	0	1	q = State prior to clock Lo to Hi Transition

FIGURE 1



Read to dev addr C0X0:: Read Status & Clear Int.
Read to dev addr C0X4:: Read Status & Compl. Int Enable
Read to dev addr C0X8:: Read Status
Read to dev addr C0XC:: Read Status & Enable Int

Status:	Int Pending	Int Enabled	60 Hz	120 Hz	float

To sniff for device: Disable interrupts, read C0X0, Read C0XC & Chk for Int disabled, Read C0XC & chk for Int Enabled, etc. & verify 60hz & 120 hz Read C0xC & C0x8 leave it pending, and Read C0X0 clears it.

The INTSRC Card

Hardware

The card consists of three chips, a 14-stage counter, a flip-flop, and a buffer chip. The card generates an interrupt every 16384 (2^14) cycles of the ø1 Clock signal by overflowing the counter. One flip-flop is used to catch the counter overflow and generate the in-

terrupt signal. The other flip-flop is used as an enable that allows the interrupt flip-flop to be set. If the enable flip-flop is clear, then the interrupt flip-flop will ignore the counter overflow. Clearing the enable flip-flop provides a dual function, it clears the interrupt flip-flop and disables further interrupts.

Reading the card has two effects. First, it returns the device status, giving the correct values of the interrupt and enable flip-flops, and also returns the value of the two high order bits of the counter. This status can be used to detect whether the card caused the interrupt. Second, reading the device may set or clear the enable flip-flop depending on the specific addresses used.

Interrupts are enabled on the card by reading the device select address $C0nC, where n is 8+slot number of the card. Interrupts are disabled and cleared by reading address $C0n0. The interrupt status can be read, without changing the enable status, by reading address $C0n8.

When the system is reset, interrupts are disabled and cleared. This card does not implement the daisy-chain protocol (although it only requires a couple gate).

INTSRC Card

Identification

The card can be identified in a variety of ways, but it must be done very carefully. All the techniques for identifying this card require reading the DEVICE SELECT address space. Doing this on other cards may activate some function on these other cards. Start the process of identifying the card by ignoring any card that has ROM in the I/O Select address space, since the INTSRC card doesn't have any ROM. This can be detected by verifying that the first 128 bytes of ROM don't change when reading them 8 times or more. This can also be done by adding up the contents of the ROM and confirming that the result is always the same.

Next, for the slots that don't have a ROM, disable the processor interrupts, and read $C0n0 to disable the card's interrupt. Read address $C0nC and make sure the interrupt enable bit (6) is clear. Reading this address will also enable interrupts, so read the address $C0nC again and verify that interrupts are enabled (status bit 6 is set).

Other conditions that can be checked include:

1) Status ($C0n8) bits 4&5 will count down (11,10,01,00,11...) and change states every 4096 cycles.
2) Status bit 7, interrupt pending, can only be set if interrupts are enabled, should remain set if $C0nC or $C0n8 is read, and should be cleared if $C0n0 is read.

The INTSRC Card

Software

The software consists of three parts, the set-up routines, the interrupt handler routines, and the shut-down routines. To keep the example simple, we have assumed that the card is in slot 3. This means that no identification routine is needed and the interrupt handler doesn't need to be reconfigured for another slot. The set-up routines install the interrupt handler into ProDOS using the ALLOC_INTERRUPT call, then enables interrupts on the card and the system processor. The interrupt handler verifies that the interrupt came from the INTSRC card. If it did, then it updates the software clock and clears the interrupt. The shut-down routines disable interrupts on the card, then remove the interrupt handler from ProDOS using the DEALLOC_INTERRUPT call.

DOS 3.3 Compatibility

While the INTSRC card was originally designed to work under ProDOS on an enhanced IIe it also works fine on an Apple II+. However, DOS 3.3 is another matter. DOS 3.3 uses the same location ($45) that the monitor interrupt handler uses. If an interrupt occurs while DOS 3.3 is using this location, then the monitor interrupt handler will overwrite it. After the interrupt is finished and control is passed back to DOS 3.3, this location will contain the wrong data, resulting in DOS 3.3 performing the wrong operation.

The Software Clock

When the card is used to provide an interrupt for a software clock, it will only be accurate if interrupts aren't locked out for a period longer than ~20 ms. However, the disk routines have this nasty habit of locking interrupts out for a longer period of time, which causes the software clock to miss interrupts and therefore lose time during disk accesses. Go ahead, make my day, format a 5 1/4" floppy disk and watch what happens.

The software example provides a simple software clock function. It counts 3737 interrupts per minute. See the calculation in Figure 2.

Finally, here's an exercise for the more advanced user. Write a driver which allows the interrupt handler to be interrupted. This is called a re-entrant driver and can be quite tricky. Any absolute location which is used by the monitor, ProDOS, or your interrupt handler must be stored in a relocatable memory area such as the system stack or an application defined stack. Otherwise the second interrupt will overwrite the data from the first interrupt.

While we realize all of you are true hardware hackers and will build your card from the schematic shown in Figure 1, you may have some friends who aren't quite sure which end of the soldering iron to pick up (a sure sign of a software hacker). Save them the trouble and buy them a card for $29 from:

PBI Software, Inc.
1163 Triton Drive
Foster City, CA.94404
(415) 349-8765

For further information on interrupt hardware and software rules and

FIGURE 2

$$14318180 \frac{\text{cycles}}{\text{seconds}} \times \frac{1 \text{ 01 cycles}}{14 \text{ cycles}} \times \frac{1 \text{ Interrupt}}{16384 \text{ 01 cycle}} \times \frac{60 \text{ second}}{1 \text{ minute}} = 3745 \frac{\text{Interrupts}}{\text{minute}}$$

However, there aren't exactly 14 cycles in a 01- cycle. At the end of every video line the 01-cycle is stretched a little:

$$\frac{455}{456} \times 3745 = 3737 \text{ Interrupts per minute}$$

requirments see:
ProDOS Technical Reference Manual. Apple IIe, IIc, and IIGS Technical Reference Manuals.

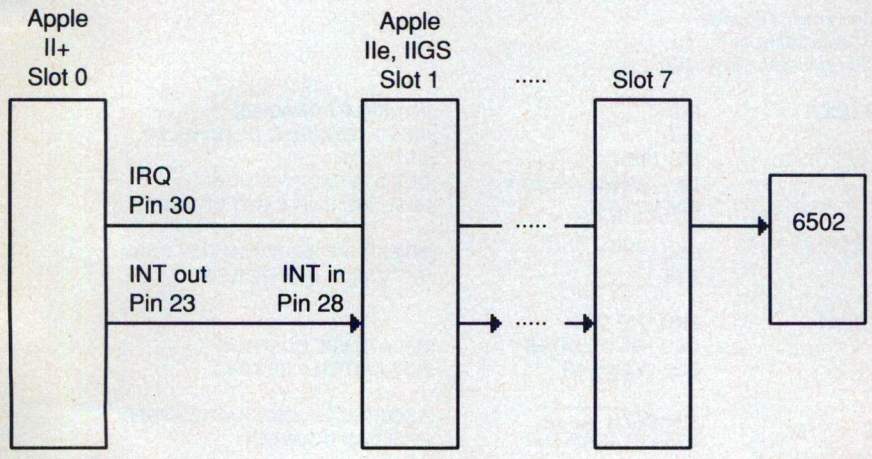
The Daisy-Chain Protocol

The daisy-chain protocol was designed to allow interrupt devices to arbitrate their priority in hardware. However, the actual implementation of this protocol leaves much to be desired. First, the protocol only allows a card to disable another card in a higher number slot. Since in many cases I/O cards are restricted to certain slots for software compatibility reasons this constraint limits the daisy-chain usefulness. Second, the protocol only works if all the slots between the two interrupt cards are filled with cards that follow the protocol. A simple solution to this problem is to make sure that the cards are next door to each other. Finally, the protocol does not necessarily reduce the time it takes to recognize and process interrupts. Since ProDOS searches for the interrupting device in the order that the interrupt handlers were installed, if the interrupt handler for the higher numbered card was installed first, then it will still be called before the lower slot's handler. This means that while the card in the lower slot may prevent the higher numbered card from interrupting, the interrupt handler overhead of the higher card must still be processed.

The protocol structure works as follows. An interrupt card may pull IRQ low only if the INT IN line for the card's slot is high. If the INT IN line is low, because some other card in a lower slot has generated an interrupt, then the card in the higher slot may not generate an interrupt. If a card generates an interrupt and pulls the IRQ line low then it should also pull the INT OUT line low to disable interrupts from cards in higher numbered slots. Otherwise, the card should pass the INT IN signal onto the INT OUT line to make sure that cards in higher numbered slots will see the interrupt priority from a card in a lower slot.

One last note on the daisy chain protocol. In one of Apple's Technical Notes they suggest generating the IRQ signal by just 'ANDing' the INT IN signal with the interrupt signal from the peripheral device. However, this is not sufficient. The state of the INT IN signal should also be considered when the interrupt device is interrogated by the interrupt handler. If the interrupt from a device is being held off by the INT IN signal from a lower slot, then it should not indicate that it has generated an interrupt when it is polled by the interrupt handler. Otherwise, ProDOS will just assume it was the interrupting device anyway, which defeats the purpose of the interrupt priority structure.

Daisy - Chain Diagram



Please rate this article on the Reader Service Card by circling:

61 Excellent **62** Good **63** Fair **64** Poor

NEW FOR APPLE //GS

Diversi-Key™
Keyboard Macro Program

Boot **Diversi-Key** once at power-on, and add macro control to **ALL** your programs! One macro keystroke can type in thousands of pre-programmed keys for you.

Call up to 100,000 macros in 320K memory! Define, edit, display, load, or save your macros at any time. Powerful programming options include nested macro calls, repeat, if/while, counter macro, time/date, and keyboard input.

Add mouse control to any program! Each mouse movement triggers one of 10 user-defined mouse macros.

Works with ProDOS, Pascal 1.3, DOS 3.3. Full set of Appleworks macros included.

Includes **FREE Diversi-Hack™**, a programmer's dream! Lets you enter the monitor any time, then return to your program.

For Apple //gs with 512K minimum **$45**

Diversi-CACHE™
Apple Disk 3.5 Speed-up

Diversi-Cache brings your new Apple Disk 3.5 up to Ramdisk speed, without Ramdisk hassle. Just boot **Diversi-Cache** once at power-on, and watch your Disk 3.5 programs run up to 5 times faster! Works with ProDOS, Pascal 1.3, DOS 3.3.

Boot times for Apple //gs System Disk	
Without **Diversi-CACHE**	41.0 sec
With **Diversi-CACHE**	19.2 sec
Reboot With **Diversi-CACHE** ..	9.6 sec

Speeds up "dumb" 3.5 drive only.
For Apple //gs with 512K minimum **$35**
Includes **FREE Diversi-HACK**

Diversi-COPY™
Fast 3.5 & 5.25 Back-up

Now you can copy or mass-produce unprotected 3.5 or 5.25 disks at maximum speed. Uses up to 800K memory in the //gs, //e aux. slot RAM card, or Apple RAM card for 1-drive, 1-pass copying.

"It's indispensable" - inCider magazine Editors' Choice for Jan. 1987.

For Apple //gs, //c, //e, II+ **$30**

Also from DSR (For //gs, //e, //c, II+)
Diversi-DOS™ **$30**
Top-rated DOS 3.3 Enhancement.
Diversi-DIAL™ **$475**
7-line BBS/Chat Demo $5

24-Hour Toll-Free Ordering
800 835-2246 ext. 127 (orders only).
For information call 313 553-9460.

Sold by mail-order only. Price includes 1st-class or foreign airmail. Return within 30 days for full refund if not fully satisfied. Visa/Mastercard (with exp. date), Check, Purchase orders accepted.

Diversified Software Research, Inc.
34880 Bunker Hill
Farmington, MI 48018-2728

CIRCLE **8** ON READER SERVICE CARD

LISTING 1

```
0000:                          1 *  Code for JEEVES Interrupt Source (INTSRC) Card
0000:                          2 *  Copyright 1987 Ninth Wave  Technologies
0000:                          3 *
0000:              C080        4 DEVSEL          EQU $C080          ;DEVICE  SELECT
0000:              0000        5 DISABLE         EQU 0              ;DISABLE INTERRUPT ADDRESS
0000:              0008        6 READ            EQU 8              ;READ INTERRUPT STATUS ADDRESS
0000:              000C        7 ENABLE          EQU $C             ;ENABLE INTERRUPT ADDRESS
0000:              BF00        8 MLI             EQU $BF00
0000:                          9 *
---- NEXT OBJECT FILE NAME IS INTSRC.0
0300:              0300       10                 ORG $300
0300:                         11 *
0300:4C  04  03               12 BEGIN           JMP ALLOC.INT
0303:30                       13 SLOT            DFB $30            ;ASSUME SLOT IS 3 FOR NOW
0304:                         14 *
0304:                         15 *  ALLOCATE INTERRUPT AND TURN ON CARD
0304:                         16 *
0304:20  00  BF               17 ALLOC.INT       JSR MLI            ;CALL MLI
0307:40                       18                 DFB $40            ; ALLOCATE INTERRUPT
0308:C7  03                   19                 DW AIPARMS         ;PARAMETER TABLE
030A:D0  08        0314       20                 BNE ERR            ;GOT ERROR
030C:                         21 *
030C:                         22 *  Enable interrupts by setting enable flip-flop
030C:                         23 *
030C:              030C       24 ENABLE.INT      EQU *              ;ENABLE INTERRUPTS
030C:AC  03  03               25                 LDY SLOT           ;GET SLOT
030F:B9  8C  C0               26                 LDA DEVSEL+ENABLE,Y ;SLOT ENABLE INTRPT ADRS
0312:58                       27                 CLI                ;ENABLE INTERRUPTS ON 6502
0313:60                       28                 RTS
0314:                         29 *
0314:00                       30 ERR             BRK                ;BREAK ON ERRORS
0315:                         31 *
0315:                         32 *  Disable interrupts by clearing enable flip-flop
0315:                         33 *
0315:              0315       34 DISABLE.INT     EQU *              ;DISABLE INTERRUPTS
0315:AC  03  03               35                 LDY SLOT           ;GET SLOT
0318:B9  80  C0               36                 LDA DEVSEL+DISABLE,Y ;SLOT DISABLE INTRPT ADRS
031B:                         37 *
031B:              031B       38 DEALLOC.INT     EQU *
031B:AD  C9  03               39                 LDA AIPARMS+1      ;GET INT.NUM
031E:8D  CC  03               40                 STA DIPARMS+1      ;AND USE TO DEALLOCATE
0321:20  00  BF               41                 JSR MLI
0324:41                       42                 DFB $41            ;DEALLOCATE INTERRUPTS
0325:CB  03                   43                 DW DIPARMS         ;PARAMETER TABLE
0327:D0  EB        0314       44                 BNE ERR
0329:60                       45                 RTS
032A:                         46 *
032A:                         47 *  Routine to handle the interrupts
032A:                         48 *  Starts by verifying that INTSRC card caused interrupt
032A:                         49 *  If not caused by INTSRC card ->returns w/ carry set
032A:                         50 *
032A:                         51 *  Increments counter
032A:                         52 *  Clears Interrupt
032A:                         53 *  Returns with Carry clear
032A:                         54 *
032A:              032A       55 INT.HDLR        EQU *              ;INTERRUPT HANDLER
032A:D8                       56                 CLD                ;INDICATES START OF HANDLER
032B:AC  03  03               57                 LDY SLOT           ;GET SLOT
032E:B9  88  C0               58                 LDA DEVSEL+READ,Y  ;GET INTERRUPT STATUS
0331:30  02        0335       59                 BMI GOT.INT        ;HI BIT INDICATES GOT INTRUPT
0333:                         60 *
0333:38                       61                 SEC                ;INTERRUPT NOT FROM THIS CARD
0334:60                       62                 RTS                ;SET CARRY AND RETURN
0335:                         63 *
0335:              0335       64 GOT.INT         EQU *
0335:CE  C6  03               65                 DEC VBL.COUNTER    ;UPDATE VBL COUNTER
0338:D0  73        03AD       66                 BNE CLEAR.INT      ;NOT ANOTHER SECOND
033A:                         67 *
033A:A9  3F                   68                 LDA #63            ;ANOTHER SECOND HAS ELAPSED
033C:8D  C6  03               69                 STA VBL.COUNTER    ;PRELOAD COUNTER
033F:                         70 *
033F:EE  C1  03               71                 INC SECONDS1
0342:AD  C1  03               72                 LDA SECONDS1       ;GET SECONDS
0345:C9  0A                   73                 CMP #10
0347:90  3A        0383       74                 BCC DISPLAY
0349:A9  00                   75                 LDA #0
034B:8D  C1  03               76                 STA SECONDS1
```

```
034E:EE  C2 03                  INC SECONDS10
0351:AD  C2 03                  LDA SECONDS10
0354:C9  06                     CMP #6                  ;10'S OF SECONDS
0356:90  2B      0383           BCC DISPLAY             ;NOT A NEW MINUTE
0358:A9  00                     LDA #0                  ;RESET SECONDS
035A:8D  C2 03                  STA SECONDS10
035D:A9  14                     LDA #20                 ;PRELOAD COUNTER TO FUDGE
035F:8D  C6 03                  STA VBL.COUNTER         ; FOR EVERY MINUTE
0362:EE  C3 03                  INC MINUTES1            ;UPDATE MINUTES
0365:AD  C3 03                  LDA MINUTES1
0368:C9  0A                     CMP #10
036A:90  17      0383           BCC DISPLAY
036C:A9  00                     LDA #0
036E:8D  C3 03                  STA MINUTES1
0371:EE  C4 03                  INC MINUTES10
0374:AD  C4 03                  LDA MINUTES10
0377:C9  06                     CMP #6
0379:90  08      0383           BCC DISPLAY             ;NOT A NEW HOUR
037B:A9  00                     LDA #0
037D:8D  C4 03                  STA MINUTES10           ;RESET MINUTES
0380:EE  C5 03                  INC HOURS1              ;UPDATE HOURS
0383:                   *
0383:            0383   DISPLAY  EQU *
0383:A2  00                     LDX #0
0385:AD  C5 03                  LDA HOURS1
0388:20  B8 03                  JSR BLASTCHAR           ;PRINT DIRECTLY ONTO SCREEN
038B:A9  3A                     LDA #':'
038D:20  B8 03                  JSR BLASTCHAR
0390:AD  C4 03                  LDA MINUTES10
0393:20  B8 03                  JSR BLASTCHAR
0396:AD  C3 03                  LDA MINUTES1
0399:20  B8 03                  JSR BLASTCHAR
039C:A9  3A                     LDA #':'
039E:20  B8 03                  JSR BLASTCHAR
03A1:AD  C2 03                  LDA SECONDS10
03A4:20  B8 03                  JSR BLASTCHAR
03A7:AD  C1 03                  LDA SECONDS1
03AA:20  B8 03                  JSR BLASTCHAR
03AD:                   *
03AD:            03AD   CLEAR.INT EQU *
03AD:AC  03 03                  LDY SLOT
03B0:B9  80 C0                  LDA DEVSEL+DISABLE,Y    ;CLEAR INTRUPT BY DISABLE
03B3:B9  8C C0                  LDA DEVSEL+ENABLE,Y     ;THEN ENABLE INTERRUPT
03B6:                   *
03B6:            03B6   EXIT     EQU *
03B6:18                         CLC                     ;INDICATE INTERRUPT PROCESSED
03B7:60                         RTS
03B8:                   *
03B8:                   *  Blast character directly onto screen
03B8:                   *  OF COURSE YOU'LL NEVER DO THIS SINCE IT VIOLATES EVERY APPLE GUIDELINE
03B8:                   *  but since this is just a simple demo we're cheating.
03B8:                   *
03B8:29  0F             BLASTCHAR AND #$0F              ;GET CHAR
03BA:09  B0                     ORA #$B0                ;ADD ASCII
03BC:9D  F1 07                  STA $7F1,X              ;BOTTOM RIGHT CORNER OF SCREEN
03BF:E8                         INX                     ;next char
03C0:60                         RTS
03C1:                   *
03C1:00                 SECONDS1 DFB 0
03C2:00                 SECONDS10 DFB 0
03C3:00                 MINUTES1 DFB 0
03C4:00                 MINUTES10 DFB 0
03C5:00                 HOURS1   DFB 0
03C6:3F                 VBL.COUNTER DFB 63
03C7:                   *
03C7:02  00             AIPARMS  DFB 2,0                ;2 PARAMETERS & INT.NUM
03C9:2A  03                     DW INT.HDLR             ;POINTER TO INTERRUPT HANDLER
03CB:01  00             DIPARMS  DFB 1,0                ;1 PARAMETER & INT.NUM
03CD:                   *
    03C7 AIPARMS            0304 ALLOC.INT          ?0300 BEGIN          03B8 BLASTCHAR
    03AD CLEAR.INT          ?031B DEALLOC.INT       C080 DEVSEL          03CB DIPARMS
    00 DISABLE              ?0315 DISABLE.INT       0383 DISPLAY         0C ENABLE
    ?030C ENABLE.INT        0314 ERR                ?03B6 EXIT           0335 GOT.INT
    03C5 HOURS1             032A INT.HDLR           03C3 MINUTES1        03C4 MINUTES10
    BF00 MLI        08 READ 03C2 SECONDS10          03C1 SECONDS1
    0303 SLOT               03C6 VBL.COUNTER
** SUCCESSFUL ASSEMBLY := NO ERRORS
** ASSEMBLER CREATED ON 15-JAN-84 21:28
** TOTAL LINES ASSEMBLED 145
** FREE SPACE PAGE COUNT 83
```

Modular Assembly Language Programming, Part V: The Apple IIGS

Bob Bishop (The PACK-Man)

This month, we will take a brief look at the Apple IIGS and see how our concepts of modular assembly language programming apply to it.

Pre-Conceived Notions

When the Apple IIGS was first introduced, information about the machine was rather vague and sketchy. What little information I could find, came mainly from a few magazine articles and brochures. They indicated that the machine could run at two different speeds: normal (the same speed as a regular Apple II), and fast (approximately two and a half times faster than a regular Apple II). Furthermore, the literature mentioned a custom chip (the Mega II) and two different modes in which the computer could operate: Emulation Mode (in which the machine looked like an Apple II), and Native Mode (in which the full capabilities of the hardware became available).

Since I had not yet actually seen the new computer, I tried to piece together the information that I had gathered and formed a preliminary concept of what the machine was probably going to be like:

I deduced that the Apple IIGS was, in effect, two distinct computers in one. When running in Emulation Mode, the computer looked like an Apple II and ran at the normal speed. When running in Native Mode, the computer ran faster, had a new set of machine language instructions, and had the new features of Super Hi-Res graphics and improved sound capability. Furthermore, the new features of the machine would only work in Native Mode and were "locked out" when running in Emulation Mode. (After all, Emulation Mode was supposed to imitate a standard Apple II, and the old Apple IIs didn't have such exotic features!)

When I finally got my Apple IIGS, I found out that many of my preconceived notions had been wrong. For one thing, the speed at which the computer runs is independent of whether it's in Emulation Mode or Native Mode. In fact, the terms "Emulation Mode" and "Native Mode" themselves are somewhat misleading.

Emulation Mode vs. Native Mode

The terms "Emulation Mode" and "Native Mode" have become common buzz-words in Apple IIGS circles, especially among users who don't know how to program the machine. When such a user sees a new program running on the computer, he usually asks a question like, "Is this program running in Native Mode?" He seems to have gotten the idea that programs running in Native Mode are somehow "superior" to those that run in Emulation Mode. But is this really true? Just what is the difference between these two modes? To answer these questions, we need to understand a little bit about the architecture of the 65C816, the microprocessor used in the Apple IIGS.

In a nutshell, the 65C816 microprocessor is very similar to the 6502 microprocessor, except that it runs faster, has a few more op-codes and addressing modes, and has bigger registers. In addi-

tion, the Processor Status Register contains several new flags. One of these flags is the Emulation (e) Flag. If this flag is set to "1", the 65C816 will act like a 6502 (with 8-bit registers). This is the so called "Emulation Mode". To switch to Native Mode, one simply has to clear this flag. Doing so, however, does not make the computer run any faster nor does it increase the size of your registers. (But, you can now tell all your friends that your program is running in "Native Mode!") Merely switching into Native Mode buys your program very little.

Once you are in Native Mode, you have the option of incorporating certain features that are not available to you when running in Emulation Mode. For example, in Native Mode you can choose the size of your index registers to be either 8 bits or 16 bits. Similarly, you can have either an 8-bit or 16-bit Accumulator register. You can even mix 'n' match different register sizes (i.e., such as having 16-bit index registers with an 8-bit Accumulator). The sizes of these registers are selected by setting or clearing the m and x flags in the Processor Status Register.

Since the e, m, and x flags are freely accessible, a program can easily switch back and forth between the different modes as often as it wants to. For example, a subroutine might start out in Emulation Mode, switch to Native Mode with 8-bit index registers and a 16-bit Accumulator, then switch to 16-bit registers with a 16-bit Accumulator, and terminate by switching back to Emulation Mode!

If all this sounds like it could be a little confusing, you're right! It becomes especially confusing when you enter the Monitor and try to disassemble a block of code containing immediate references to memory (such as LDA #0). Depending on whether the register to be loaded is currently 8 bits long or 16 bits long, the disassembly will show either a 2-byte instruction or a 3-byte instruction. Since the size of the registers are subject to dynamic change, it is almost impossible to guarantee that the code shown in such a disassembly is actually the code being executed! For example, you may have intended for your program to use 8-bit registers. But if the flags specifying the register sizes are incorrectly set, a load immediate instruction will actually grab the op-code portion of the next instruction of your program and use it as a byte of data!

Programming the 65C816 can be a little tricky, especially for a beginning programmer. If you plan to program the Apple IIGS in assembly language, I refer you to any of the books available on the subject, such as *Programming the 65816*, by David Eyes/Ron Lichty, published by Prentice Hall Press. In the discussion that follows, I will assume that you have already developed at least a working understanding of the 65C816 microprocessor.

Using APW to Program the Apple IIGS

Since the Apple IIGS was designed to be compatible with the older series of Apple IIs, you can program it in pretty much the same way as you've been programming the Apple II. The Apple IIGS even contains the old Applesoft BASIC in its ROMs. But if you plan to do professional software development, the Apple IIGS Programmers Workshop (APW) is available from the Apple Programmer's and Developer's Association (APDA). APW is a development environment for the Apple IIGS which consists of the following components:

- The Shell (the interface that allows you to execute the desired APW programs).
- Text Editor (for creating and editing source code).
- 65C816 Assembler (for generating 65C816 object code).
- C Compiler (for programming in the C language) Not available yet.
- Linker (for creating load files out of object files).
- Debugger (to facilitate the debugging of assembly language programs).
- Utility Programs (for performing miscellaneous functions not built into the Shell).

This development system represents a nice integrated package in that all of its constituent pieces are designed to work in harmony with each other. Programs can be written in the C language, or in assembly language, or in a combination of the two. This is because all of the programming languages in APW generate object files that are compatible with each other.

However, to use APW, your Apple IIGS must be equipped with two 3.5 microfloppy disk drives as well as a 1Mb memory expansion card. Furthermore, APW is so big and powerful, it comes with several pounds of documentation for you to read.

But if you've just gotten your Apple IIGS and are impatient to start writing programs or you didn't buy the 1Mb memory expansion card, or you don't have two disk drives, or you don't like ProDOS (APW works only under ProDOS) or if you have a hard disk containing years of DOS 3.3 work that you will lose if you convert to ProDOS, there is another alternative. The assembler you are now using on the Apple II will probably work just fine on the Apple IIGS. Furthermore, you can use it to generate "Native Code", even if your assembler was not written for the 65C816. To illustrate, let's see how EDASM can be used to generate code for the Apple IIGS.

Using EDASM To Program The Apple IIGS

From a programmer's point of view, the 65C816 is not a totally new microprocessor; it is merely an enhanced 6502. This means that, even though it contains several new features, it still has all the attributes of a 6502. Not only does it contain all of the 6502 mnemonics (even in Native Mode), but those mnemonics assemble into their same hexadecimal op-codes (i.e., LDA immediate is still $A9, JSR is still $20, etc.). Therefore, EDASM (or any other 6502 assembler) will handle all of these old 6502-type instructions just fine.

But how do we handle the new instructions that didn't exist in the 6502? Simply define them by using an EQU pseudo-op, and then implement them with a DFB pseudo-op. For example, if we wanted to write a subroutine that runs in full Native Mode (16-bit Accumulator and 16-bit registers), we could structure it as follows:

```
START   CLC
        DFB     XCE
        PHP
        DFB     REP,$30

etc.

RETURN  PLP
        DFB     XCE
        RTS
*
REP     EQU     $C2
XCE     EQU     $FB
```

Almost all of the new instructions in the 65C816 microprocessor can be handled in this manner. You can even implement the "long" instructions. For example, to load the Accumulator from memory location $123456, simply use an EQU pseudo-op to declare LDAL to

have the hexadecimal value $AF, and then use a:

```
DFB     LDAL,$56,$34,$12
```

at the point where you wish the operation to occur.

When using 16-bit registers you must be careful how you implement instructions that use the immediate mode. Since EDASM expects only an 8-bit operand, you must remember to follow such instructions with a DFB psuedo-op to supply the second byte of the operand. For example, to load the 16-bit value $1234 into the Accumulator, you would need to:

```
LDA     #$34
DFB     $12
```

All of this may appear to be somewhat awkward, but it's really not so bad when you get used to it. Fortunately, the bulk of a typical 65C816 program consists mainly of 6502-type instructions, so the problem isn't quite as bad as it may first appear.

Designing Modular Code For The Apple IIGS

In the December, 1986 issue of *Call-A.P.P.L.E.* magazine, I presented a set of guidelines for writing assembly language programs. (See "Lets Start Publishing Usable Assembly Language Code", pages 21-23 of that issue.) In particular, one of those guidelines (Rule 2) stated:

"Avoid Confiscating Any of the Computer's Resources"

In other words, subroutines should be written in such a manner that they avoid "stepping on the toes" of any other subroutines that might also be in the machine at the same time. Indeed, the design of such subroutines has been the subject of this monthly column.

Rule 2 was actually the embodiment of the following four sub-rules:

a. Maintain the Integrity of the X and Y Registers.
b. Maintain the Integrity of Page Zero.
c. Avoid Using Absolute Memory Locations For Passing Subroutine Calling Parameters.
d. Write Relocatable Subroutines.

All of the these guidelines are still valid for the Apple IIGS. But, because of its more complicated architecture, we need to include a few additional guidelines:

Because the 65C816 microprocessor can run either in Emulation Mode or Native Mode, a subroutine should remember to:

e. Maintain The Running Mode Of The Microprocessor.

Furthermore, since the Native Mode of the 65C816 allows for two different size index registers (8-bit or 16-bit), a subroutine should:

f. Maintain The Size Of The X And Y Registers.

Similarly, since the size of the Accumulator is also subject to change, subroutines should:

g. Maintain The Size Of The Accumulator Register.

Your subroutine can, of course, temporarily change the state of the computer to be whatever it needs (8-bit or 16-bit registers, Native or Emulation Mode), as long as the subroutine remembers to re-establish the state that existed when it was invoked.

All of these considerations suggest the following "Cookbook" interface for subroutines that are to run in Native Mode on the Apple IIGS:

1. Save the contents of the X and/or Y index registers (if your subroutine requires the use of those registers).
2. Set the processor to Native Mode (CLC followed by XCE).
3. Push the processor status register onto the stack (so that it can later restore the m,x, and e flags).
4. Set the registers to the sizes you want them to be (by using REP and/or SEP instructions).
5. Let your subroutine "do its thing".
6. Pull the Processor Status Register off of the stack (re-establishing the former sizes of the index registers and the Accumulator).
7. Perform an XCE (to re-establish the former running mode of the microprocessor).
8. Restore the former contents of the X and/or Y index registers (if they were used).
9. RTS back to the calling program.

An Example: Super Hi-Res Plotting

As an example of everything we have discussed so far, consider the subroutine module shown in Listing 1. This subroutine plots a pixel on the 320x200 Super Hi-Res graphics screen. (It assumes the screen has already been initialized and that a color palette has been installed.)

Lines 19-23 reflect the first four steps of the "Cookbook" interface. In lines 25-49, the subroutine "Does its own thing." Finally, in lines 53-56, the subroutine performs Steps 6 through 9 of the "Cookbook" interface. Notice that lines 62-68 contain the EQUates that define the new 65C816 instructions used by the subroutine.

Super Hi-Res graphics is one of the most prominent features that has been added to the Apple IIGS. With it you can plot in either 320x200 or 640x200 resolution. In addition, you have 4096 colors from which to choose, although the actual number of different colors that can easily be displayed at one time is limited to 256. (But with "tricky" programming you can display several thousand.)

Unfortunatly, space does not permit us to discuss the Super Hi-Res graphics mode at length. Perhaps at a later date we will once again return to this interesting feature of the Apple IIGS.

Using Page Zero

One of the problems with the 6502 was that everyone wanted to use the memory locations on Page Zero but there just weren't enough to go around. So acute was the problem that one of our previous guidelines explicitly stated:

Maintain the integrity of Page Zero.

On the Apple IIGS, this problem has been virtually eliminated. As we all know, the 65C816 has a user-definable Page Zero. It is now called the Direct Page and can be located anywhere in bank $00. (We will use the terms "Page Zero" and "Direct Page" interchangable.)

But making its location user-definable is *not* the reason why Page Zero has become less of a problem. The real reason is because we no longer *need* to use Page Zero as often.

In the 6502, with its 8-bit index registers, it was difficult to access large blocks of memory (such as the Hi-Res graphics buffer). The only way to do it, other than by writing self-modifying code, was to establish a 2-byte pointer on Page Zero. This pointer could then be set to point anywhere in memory.

The 65C816, on the other hand, provides us with the option of having 16-bit index registers. Therefore, we no

longer need pointers on Page Zero to access large blocks of memory. (I have been programming my Apple IIGS for several months and have written dozens of subroutine modules, and I have yet to need any memory locations on the Direct Page.)

However, if you do write applications that use Page Zero, don't let the fact that Page Zero is user-definable lull you into thinking that you are free to use it at will. User-definable or not, the Direct Page represents a chunk of memory, and rule 2 warns us against confiscating such resources.

The Apple IIGS Toolbox

Finally, we come to that part of the Apple IIGS that bears a strong kindred spirit to the philosophies being established in this monthly column. The Toolbox represents Apple's attempt at supplying programmers with a collection of subroutine modules to make the task of programming easier.

Although our overall philosophies are in complete agreement, Apple chose to implement their Toolbox subroutines with slightly different guidelines than the ones we have been using. Let's take a brief look at the similarities and differences between the two approaches.

How To Use The Toolbox

The Toolbox is actually a collection of separate Tool Sets. A Tool Set in the Apple IIGS is a collection of related subroutines that all deal with a common theme, much like a P.A.C.K. library does. For example, the Memory Manager Tool Set allocates memory to your application, the QuickDraw Tool Set contains subroutines for handling graphics, etc.

The subroutines in the Toolbox are referenced, not by an ENTRY name (as we have been doing), but by a 16-bit "code number." The user loads the X register with this code number and performs a JSL to memory location $E10000. If the subroutine has any calling parameters, they are passed on the stack. Upon returning, the stack contains the subroutine's output parameters (if any). If an error occurred, the carry flag is set and the Accumulator contains an error code.

Each subroutine in the Toolbox is invoked in this same straightforward way. By using the stack for passing its calling parameters the subroutine avoids confiscating absolute memory locations. This is in complete agreement with our Rule 2. But Rule 2 also warns us

LISTING 1

```
Source      File:

1000:          1000                ORG     $1000
1000:                              REL
1000:                      *
1000:          1006                ENTRY   UHPLOT
1000:                      *
1000:                      * PLOT A PIXEL(320X200 GRAPHICS)
1000:                      *
1000:                      * ENTER WITH:
1000:                      *     UHPLOT-2: 2-BYTE XCOORDINATE
1000:                      *     UHPLOT-4: 2-BYTE Y-COORDINATE
1000:                      *     UHPLOT-6: 2-BYTE COLOR(MUST
1000:                      *           BE IN RANGE: 0- 15)
1000:                      *
1000:          0001        ZZ          DS      1
1001:00                            DFB     $00
1002:          0002        YY          DS      2
1004:          0002        XX          DS      2
1006:                      *
1006:DA                    UHPLOT      DFB     PHX
1007:18                            CLC
1008:F8                            DFB     XCE
1009:08                            PHP
100A:C2 30                         DFB     REP,$30
100C:                      *
100C:AD 01 10                      LDA     YY-1
100F:4A                            LSR     A
1010:4A                            LSR     A
1011:6D 01 10                      ADC     YY-1
1014:6D 04 10                      ADC     XX
1017:4A                            LSR     A
1018:AA                            TAX
1019:E2 20                         DFB     SEP,$20
101B:B0 16     1033                BCS     ODD
101D:                      *
101D:AD 00 10              EVEN        LDA     ZZ
1020:0A                            ASL     A
1021:0A                            ASL     A
1022:0A                            ASL     A
1023:0A                            ASL     A
1024:8D 44 10                      STA     COLOR
1027:BF 00 20 E1                   DFB     LDAL.X,$00,$20,$E1
102B:29 0F                         AND     #$0F
102D:0D 44 10                      ORA     COLOR
1030:4C 3C 10                      JMP     DONE
1033:                      *
1033:BF 00 20 E1           ODD         DFB     LDAL.X,$00,$20,$E1
1037:29 F0                         AND     #$F0
1039:0D 00 10                      ORA     ZZ
103C:9F 00 20 E1           DONE        DFB     STAL.X,$00,$20,$E1
1040:                      *
1040:                      * DONE
1040:                      *
1040:28                            PLP
1041:FB                            DFB     XCE
1042:FA                            DFB     PLX
1043:60                            RTS
1044:                      *
1044:          0001        COLOR       DS      1
1045:                      *
1045:                      * EQUATES FOR 65C816
1045:                      *
1045:          00BF        LDAL.X      EQU     $BF
1045:          00DA        PHX         EQU     $DA
1045:          00FA        PLX         EQU     $FA
1045:          00C2        REP         EQU     $C2
1045:          00E2        SEP         EQU     $E2
1045:          009F        STAL.X      EQU     $9F
1045:          00FB        XCE         EQU     $FB
1045:                      *

**Successful assembly := No Errors
**Assembler created on 21 May 83 Rel 07
**Total Lines Assembled 70
**Free Space Page Count 78
```

against confiscating *any* of the computer's resources, including the index registers. By using the X index register for specifying the code number for a subroutine, the Toolbox has partially violated Rule 2. Furthermore, the subroutines in the Toolbox seem to make no attempt at preserving the contents of the Y index register either.

The Toolbox subroutines cannot be called from Applesoft BASIC (since BASIC cannot directly access the stack for passing the calling parameters). The approach that we have been using in this series (passing calling parameters in the memory locations immediately preceding the subroutine) allows the calling parameters to be PEEKed and POKEd from Applesoft BASIC.

However, even if the problem of how to pass calling parameters had been handled better, the Toolbox subroutines suffer from yet another design flaw: the tools are designed to run only in full Native Mode. This means that the e, m, and x Flags must all be set to zero, so that the Accumulator and index registers are all 16 bits. Since Applesoft BASIC runs only in Emulation Mode, the Tool Sets could still not be called from BASIC.

It is, of course, possible to clean up these problems by writing our own separate interface to each Tool Set subroutine. An example of this is shown in Listing 2. This example (subroutine MOVEPEN) provides a better interface to the MOVETO subroutine contained in the QuickDraw Tool Set. Notice the similarity in structure between MOVE PEN and the UHPLOT subroutine of Listing 1. Again, we start out by preserving our calling environment (lines 14-19). Next, we pass the calling parameters to the stack (lines 21-24), load the X register with the code number for the MOVETO subroutine (lines 25-26), and then perform a JSL to memory location $E10000 (in line 27). Finally, we restore the calling environment and return to the calling program (in lines 29-33).

Subroutine MOVEPEN has the advantage that it can be called from either Native Mode or Emulation Mode. Furthermore, its two calling parameters can be PEEKed and POKEd from a BASIC program. In addition, both the X and Y registers are preserved to avoid possible register conflicts.

(Apple wants to avoid people using Applesoft to connect to the Toolbox).

LISTING 2

```
Source        File:

1000:          1000   1          ORG     $1000
1000:                 2          REL
1000:                 3  *
1000:          1004   4          ENTRY   MOVEPEN
1000:                 5  *
1000:                 6  * THIS SUBROUTINE PROVIDES A
1000:                 7  * CLEANER INTERFACE TO THE
1000:                 8  * "MOVETO" SUBROUTINE OF THE
1000:                 9  * QUICKDRAW II TOOL SET
1000:                10  *
1000:          0002  11  YY       DS      2
1000:          0002  12  XX       DS      2
1004:                13  *
1004:DA               14  MOVEPEN  DFB     PHX
1005:5A               15          DFB     PHY
1006:18               16          CLC
1007:FB               17          DFB     XCE
1008:08               18          PHP
1009:C2 30            19          DFB     REP,$30
100B:                 20  *
100B:AD 00 10         21          LDA     YY
100E:48               22          PHA
100F:AD 02 10         23          LDA     XX
1012:48               24          PHA
1013:A2 04            25          LDX     #$04
1015:3A               26          DFB     $3A
1016:22 00            27          DW      CALL,TOOL
101A:                 28  *
101A:28               29          PLP
101B:FB               30          DFB     XCE
101C:7A               31          DFB     PLY
101D:FA               32          DFB     PLX
101E:60               33          RTS
101F:                 34  *
101F:          0022  35  CALL     EQU     $0022
101F:          E100  36  TOOL     EQU     $E100
101F:                 37  *
101F:                 38  * EQUATES FOR 65C816
101F:                 39  *
101F:          00DA  40  PHX      EQU     $DA
101F:          005A  41  PHY      EQU     $5A
101F:          00FA  42  PLX      EQU     $FA
101F:          007A  43  PLY      EQU     $7A
101F:          00C2  44  REP      EQU     $C2
101F:          00FB  45  XCE      EQU     $FB
101F:                 46  *

**Successful Assembly := No Errors
**Assembler Created on 21 May 83 Rel 07
**Total Lines Assembled 47
**Free Space Page Count 78
```

LISTING 3

```
Source        File:

9500:          9500   1          ORG     $9500
9500:                 2  *
9500:                 3  * THIS SUBROUTINE MUST BE CALLED
9500:                 4  * ONCE AT BOOT-TIME IF THE USER
9500:                 5  * PLANS TO USE ANY OF THE ROM-
9500:                 6  * BASED TOOL SETS
9500:                 7  *
9500:18               8  BOOTINIT  CLC
9501:FB               9          DFB     XCE
9502:C2 30           10          DFB     REP,$30
9504:                11  *
9504:                12  * RETURN IF NOT JUST POWERED-UP
9504:                13  *
9504:AF AA 00  E1    14          DFB     LDAL,$AA,$00,$E1
9508:D0 36     9540  15          BNE     RETURN
950A:                16  *
950A:                17  * CALL THE MISC. TOOLSET
950A:                18  * TO GENERATE A USER ID
950A:                19  *
```

```
950A:48                          20          PHA
950B:AD 5D 95                    21          LDA     S.1FFF
950E:48                          22          PHA
950F:AE 57 95                    23          LDX     GNID
9512:22 00                       24          DW      CALL,TOOL
9516:7A                          25          DFB     PLY
9517:B0 2A       9543            26          BCS     ERROR
9519:98                          27          TYA
951A:8F AA 00 E1                 28          DFB     STAL,$AA,$00,$E1
951E:                            29  *
951E:                            30  * CALL THE MEMORY MANAGER
951E:                            31  * TO CONFISCATE BANK $00
951E:                            32  *
951E:48                          33          PHA
951F:48                          34          PHA
9520:AD 59 95                    35          LDA     S.0000
9523:48                          36          PHA
9524:AD 61 95                    37          LDA     S.B800
9527:48                          38          PHA
9528:5A                          39          DFB     PHY
9529:AD 63 95                    40          LDA     S.C017
952C:48                          41          PHA
952D:AD 59 95                    42          LDA     S.0000
9530:48                          43          PHA
9531:AD 5B 95                    44          LDA     S.0800
9534:48                          45          PHA
9535:AE 53 95                    46          LDX     NH
9538:22 00                       47          DW      CALL,TOOL
953C:7A                          48          DFB     PLY
953D:7A                          49          DFB     PLY
953E:B0 03       9543            50          BCS     ERROR
9540:                            51  *
9540:38                          52  RETURN  SEC
9541:FB                          53          DFB     XCE
9542:60                          54          RTS
9543:                            55  *
9543:AD 5F 95                    56  ERROR   LDA     S.2BAD
9546:48                          57          PHA
9547:AD 59 95                    58          LDA     S.0000
954A:48                          59          PHA
954B:48                          60          PHA
954C:AE 55 95                    61          LDX     SFM
954F:22 00                       62          DW      CALL,TOOL
9553:                            63  *
9553:02 09                       64  NH      DW      $0902
9555:03 15                       65  SFM     DW      $1503
9557:03 20                       66  GNID DW $2003
9559:00 00                       67  S.0000  DW      $0000
955B:00 08                       68  S.0800  DW      $0800
955D:FF 1F                       69  S.1FFF  DW      $1FFF
955F:AD 2B                       70  S.2BAD  DW      $2BAD
9561:00 B8                       71  S.B800  DW      $B800
9563:17 C0                       72  S.C017  DW      $C017
9565:                            73
9565:            0022            74  CALL    EQU     $0022
9565:            E100            75  TOOL    EQU     $E100
9565:                            76  *
9565:                            77  * EQUATES FOR 65C816
9565:                            78  *
9565:            0022            79  JSL     EQU     $22
9565:            00AF            80  LDAL    EQU     $AF
9565:            005A            81  PHY     EQU     $5A
9565:            007A            82  PLY     EQU     $7A
9565:            00C2            83  REP     EQU     $C2
9565:            00E2            84  SEP     EQU     $E2
9565:            008F            85  STAL    EQU     $8F
9565:            00FB            86  XCE     EQU     $FB
9565:                            87  *

**Successful Assembly := No Errors
**Assembler Created on 21 May 83 Rel 07
**Total Lines Assembled 88
**Free Space Page Count 78
```

Starting Up The Tool Sets

Before you can use the subroutines in the Toolbox, you must first startup the Tool Sets that contain them. The method of starting up each Tool Set is fairly straightforward — you merely invoke a startup subroutine which is contained in the Tool Set itself. Logically, this approach would lead to a "Catch-22" condition: How can you access a Tool Set's initializing subroutine if the Tool Set itself is not initialized in the first place? The hack-solution to this dilemma was to merely fudge the rules a little.

But there is a more serious "Catch-22" problem in the design of the Apple IIGS Toolbox: Some of the Tools exist in ROM, while others have to be loaded in from disk. Therefore, in order to use the complete capabilities of the Toolbox, you must be running with the ProDOS operating system. But in Appendix A of the Toolbox documentation, under the section titled: "Writing Your Own Tool Set", the documentation explicitly states: "There are some factors which you must consider when writing your own Tool Set: Routines may not assume the presence of any operating system..." But, if you don't have an operating system present, how are you supposed to load in such Tool Sets in the first place!?

If you use APW to develop your software, you may not have given much thought to this inconsistency, since ProDOS is then automatically in the machine anyway. But if you want to run under a different operating system, such as DOS 3.3, you are limited to using only the ROM-based Tool Sets. Still, those ROMs contain a lot of good stuff. So let's see how we can access the ROM-based tools, even if we aren't using ProDOS.

Using The Tool Sets Without ProDOS

As we have seen, it is easy to use your favorite Apple II assembler for programming the Apple IIGS. As such, your assembler can run under ProDOS, DOS 3.3, or even your own custom operating system. But if you plan to access the Toolbox subroutines with such assemblers, you should be aware of at least one more small problem: The Tool Sets all use the Memory Manager to obtain their scratch memory. If you are generating software under a "non-standard" operating system (ProDOS-16 is the "standard"), the Memory Manager may be unaware of your existence and give away part of the memory that you had planned to use for your own code! Therefore, upon powering up your system, you should first execute the small piece of code shown in Listing 3. This program simply confiscates bank $00 for your use so that the Memory Manager won't assign it to another subroutine later on. All of bank $00 will then remain yours until the next time you power up.

The Roots Of The Apple IIGS

The Apple IIGS is the latest mutation in a family of computers whose roots

SALE • *Famous National Brand* • SAVE

NLQ 180
Hi-Speed Printer Sale

• 160 - 180 CPS • Near Letter Quality •
Lifetime Warranty*

Price Break Thru

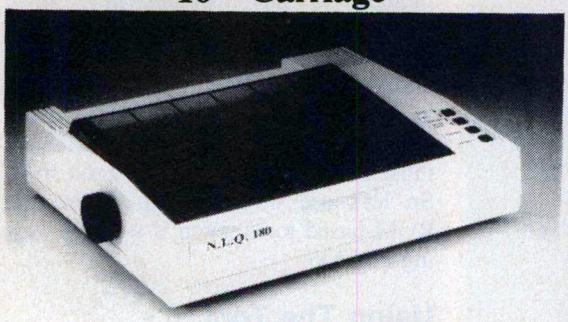
Sale $199.00

List $499.95

Below Wholesale Cost Prices!

60% OFF LIST PRICE

10" Carriage



NLQ-180 Premium Quality Printer

Near Letter Quality Selectable From Front Panel Controls • High Speed Dot Matrix• Letter Quality Modes • 8K Buffer frees up computer 4-times faster • Pica, Elite, Italics, Condensed • Super Graphics • Business or Personal • Tractor/Friction • 15 Day Free Trial • Lifetime Warranty on Print Head* • 6 Month Immediate Replacement Policy •

NEW

All New up Front Panel Controls

Fantastic Graphics

Easy to Use

Fantastic Price

Sale

NLQ-180 Print Samples

This is an example of ITALICS
Enhanced Boldface
Condensed Text Double-strike
example of Near Letter Quality

APPLE — ATARI — EPSON — NLQ180 SPECIFICATIONS — IBM — COMMODORE — ETC.

Print Buffer
8K bytes utility buffer
Printing Direction
Text Mode — Bi-directional
Graphic Mode — Uni-directional
Interface
Centronics Parallel Port
Paper
Plain paper, Roll paper, Single sheet
Fanfold, Multipart paper: max. 3 sheets (original plus 2 copies)
Character Fonts
Pica, Elite, Italics, Condensed

Printing Method
Impact dot matrix
Printing Speed
160-180 CPS at standard character printing
Printing Characters
Standard 9 x 9 dot matrix
NLQ 12 x 18 dot matrix (33cps)
Character size: 2.12 x 2.8 mm (standard)
Character sets: Full ASCII character set (96)
32 International characters

Ink Ribbon Cartridge
Ribbon Life: 3 million characters/cartridge
Physical Dimensions
Size: 15" x 12" x 5"
Weight: 12.7 lbs.
Maximum Number of Characters

Standard:	10 cpi	80 cpl
Standard enlarged:	5 cpi	40 cpl
Elite:	12 cpi	96 cpl
Elite enlarged:	6 cpi	48 cpl
Condensed:	17 cpi	132 cpl
Condensed enlarged:	8.5 cpi	66 cpl
Condensed elite:	20 cpi	160 cpl

INTERFACES

Atari $39.95 Apple II $44.95 Commodore $29.95 IBM $24.95 Laser 128 $19.95 Macintosh $49.95

Shipping, Handling & Insurance Charges
Add $10.00 for shipping, handling, and insurance. Illinois residents please add 6½% sales tax. Add $20.00 for ALASKA, CANADA, HAWAII, PUERTO RICO & APO-FPO orders. All orders must be in U.S. Dollars. WE DO NOT EXPORT TO OTHER COUNTRIES EXCEPT CANADA & PUERTO RICO. Enclose cashier check, money order or personal check. Allow 14 days for delivery, 2 to 7 days for phone orders, 1 day express mail. Prices & Availability subject to change without notice.
VISA — MASTER CARD — C.O.D. Call For C.O.D. Charges.

COMPUTER DIRECT

We Love Our Customers

22292 N. Pepper Rd., Barrington, Illinois 60010

312/382-5050 TO ORDER

CIRCLE **14** ON READER SERVICE CARD

Complete Apple® Compatible
Word Processing System

• 15 Day Free Trial • 90 Day Immediate Replacement Policy •

Home • Business • School • Word Processing

Look At All You Get for only $499.00

This computer system is sold as a complete unit, NO SUBSTITUTIONS are allowed.

		List Price	Sale Price
1	Laser 128 Apple Compatible Computer	$499.00	$395.00
2	Big Blue Printer	199.00	39.95
3	12" 35 MHz Hi-Resolution Monitor	229.95	99.00
4	Magic Window IIe wordprocessor	150.00	49.95
5	Magicalc Spreadsheet	150.00	49.95
6	Magic Memory Data Base	60.00	39.95
7	Laser 128 Big Blue Printer Interface	19.95	12.95
8	2 Rolls of Heat Transfer Paper	19.95	5.95
	Comparable Apple System ($2495.00)	$1327.85	**$692.70**

Save Over $700.00 Off List Prices

All 8 Pieces Only $499

Magic Window IIe: Word processing is easy and more efficient because of a simple-to-read menu system and editing commands. Many powerful features of professional quality programs and more. Supports virtually all major printer functions. Operates with most 80 column video cards or 40/70 column no-hardware modes.

Magicalc: Rated as the best electronic spreadsheet on the market for Apple. The speed with which Magicalc solves number problems allows you to accurately analyze decisions beforehand. Supports multiple RAM cards of most manufacturers in any combinations up to a full 512 K.

Magic Memory: File anything, any way you want to: names, addresses, important numbers, dates and notes. Your information is organized simply, easy to maintain, and available instantly. Supports all popular printers and video cards.

Apple® is the registered trademark of Apple Computers Inc.

Shipping, Handling & Insurance Charges and Information

Add $35.00 for shipping, handling and insurance. Illinois residents please add 6½% sales tax. Add $70.00 for CANADA, PUERTO RICO, HAWAII, ALASKA, and APO-FPO. WE DO NOT EXPORT TO OTHER COUNTRIES EXCEPT CANADA & PUERTO RICO. All orders must be in U.S. dollars. Enclose Cashier Check, Money Order or Personal Check. Allow 14 days for delivery, 2 to 7 days for phone orders, 1 day express mail. Prices & Availability subject to change without notice.

VISA - MASTERCARD - C.O.D. Please Call for C.O.D. Charges

COMPUTER DIRECT

22292 N. Pepper Road, Barrington Il. 60010

CALL (312) 382-5050 TO ORDER

We Love Our Customers!

CIRCLE **15** ON READER SERVICE CARD

For your Apple][e,][c, or][gs

From the Center of the Silicon Prairie

Nite Owl Developer Disks make Applesoft programming with ProDOS easy !!

- **Write Programs for entire Apple][family**
- **Not copy protected - No licensing fees**
- **Shipping included - Only $39.95 each**

Developer Disk #1 - Quick Index - Program selector, DAS Editor - super scrolling Applesoft program editor, GETSTRING, picture and text page PACKER, SETDATE, KILL command, programming tutorial

Developer Disk #2 - Directory Mapper, Easy Windows, Structured Applesoft - GOTO "Routine name", File Copy, Stick machine language onto Applesoft programs, Wow, Disk ZAP editor and mapper, alphabetize directories

---------- New Stuff ----------

Nite Owl Sampler - Demonstration Disk with things that you can use - Fast Disk][Copy, Hi-RES picture dump for ImageWriters, Packed pictures **Only $10**

The Class Selector

Designed for elementary schools K-6. Will selectively sort students into classes with equal distribution based on sex, race, reading ability,special help, or whatever. Student names are automatically alphabetized. The selector can sort 300 students per grade into a maximum of 12 classes. Student data entry is very fast and easy requiring little computer experience for the operator. Includes complete manual. Will save many hours associated with the job of arranging classes and printing rosters. Schools should rush their PO's for $195 for prompt delivery.

Shipping included - Satisfaction Guaranteed
Send personal check or money order. Use our reader service number for FREE Nite Owl News.

Nite Owl Productions
5734 Lamar
Mission, KS 66202
(913) 362-9898

CIRCLE **10** ON READER SERVICE CARD

MACRODISK™

The
Apple Writer™
Enhancement

New Version

Version **3.2** adds many new features to **MacroDisk**, the popular enhancement system for ProDOS Apple Writer. Now a complete set of ProDOS Utilities has been added to the power of 48 Apple Writer macros and over 25 WPL programs that work together to bring new commands, sophistication and speed to Apple Writer.

One Keystroke:

- Moves a block of **any** size (breaking the 1024 character limit of Apple Writer) to another position in the document, or to or from a file on disk. A document or a disk file can have any number of marked blocks able to be moved instantly.
- Catalogs, Lists Volumes, Creates a Subdirectory, Unlocks or Locks, Renames, Sets Prefix, or Deletes a File.
- Toggles Carriage Display (no more menu searching), Inserts Date at cursor, Finds text, Erases text from cursor to end of file.
- Automatically saves document to required pathname as well as saves a backup copy of the document to a different pathname!
- Prints, Previews document before printing, Prints memory to disk, or displays Print/Program Command Menu.
- Formats files for uploading, or reformats downloaded files.

MacroDisk automatically installs your printer's Control Codes and a printer menu, uploads automatically to your Ram Disk, provides a keyboard Template, a time and date Preboot, a Program Launcher, and ProDOS Utilities: Lock, Unlock or Copy or Delete files, directories, volumes. Delete forward commands, time stamp paragraphs, and much, much more.

$19.95 Money Back Guarantee

MacroDisk
2844 Lyon, Ste. 302
San Francisco, CA 94123 (415) 563-6459

CIRCLE **7** ON READER SERVICE CARD

reach back more than 10 years to the Apple-I. The standard Apple-I was based on the 6502 microprocessor, contained 4K bytes of RAM, and had a few hundred bytes of ROM. It had but one display mode: simple text.

In retrospect, the Apple-I was a relatively simply machine to program. It had no special features and, by today's standards, was not very exciting. So the following year, when the Apple II was introduced, I traded in my Apple-I for one of the first new machines.

The original Apple II was again based on the 6502 microprocessor, but now it came with 16K of RAM and several K of ROM which contained integer BASIC, a mini-assembler, and many other goodies. In addition, it had two graphics modes: Lo-Res with 16 colors, and Hi-Res with 2 colors (green and purple). Of course, these added capabilities made the computer a little more complicated to program. So the 12-page manual that originally came with the Apple-I had now expanded to well over 60 pages for the Apple II. By the time the famous "red book" came out in 1978, the documentation had jumped to more than 150 pages.

Over the years, more and more features were piled into the Apple II: the Autostart ROM was introduced, two new colors (blue and orange) were added to Hi-Res graphics, floating point BASIC was put in ROM, the size of memory kept increasing, the language card appeared, new graphics modes were introduced, etc., etc., etc. But in spite of all these changes, Apple Computer Inc. insisted on maintaining compatibility with all previous generations of the Apple II. While this was a very laudable goal, compatibility is a double-edged sword — compatibility also perpetuates the design mistakes of the past.

As a result of all of these "improvements", the Apple II has experienced severe growing pains. With all of its added-on features, the machine has become like the Winchester Mystery House. (The Winchester Mystery House is a sprawling 160-room Victorian mansion located in San Jose, not far from Apple Computer, Inc. It contains doors that open into blank walls, and stairways that dead end at the ceiling.) Even such a fundamental operation as merely accessing the computer's own memory has become a complicated mess! (Case in point: Part IV of this series presented a fairly lengthy subroutine for accessing the second 64K bank of a 128K system.) Apple's 1977 advertising slogan, "Simplicity is the ultimate sophistication", seems to have become all but forgotten.

The problems of increasing complexity are not unique to the Apple II. In their frenzy to create computers with mass appeal, manufacturers in general have bent over backwards to cater to the public's desire for "user-friendliness". Consequently, the emphasis has been on producing software and hardware that's easy to *use*. Perhaps someday an effort may be launched at designing a computer that's also easy to *program*. After all, there is no fundamental law that requires a computer system to be complicated and confusing in order for it to be powerful. But if the current trend continues, it won't be long before computers become so complicated that almost nobody will be able to program them effectively.

Conclusion

The designers of the Apple IIGS missed a golden opportunity to put an end to the growing pains suffered by the Apple II family. If a separate 6502 had been used to provide Apple II compatibility (with all the new features of the machine "locked out"), the 65C816 with its confusing multi-stated operation would not have been necessary. A second microprocessor (preferably of the 68000 type) could then have been implemented, in lieu of the Native Mode, to begin a whole new generation of Apples with a cleaned-up design. This would have provided a more powerful architecture for the new computer and put a cap on the 10-year chain of Apple II design modifications.

Instead, the current design of the Apple IIGS merely contributes a few more links to the chain of added-on features (and complexities) that exist in the already overburdened Apple II architecture. In spite of some of its attractive new features, the Apple IIGS is not a computer that I would recommend for beginning assembly language programmers. But, like any new toy, it's still kind of fun to play with — at least until the novelty wears off.

For more information about APDA:
CIRCLE **70** ON READER SERVICE CARD

Please rate this article on the Reader Service Card by circling:
71 Excellent **72** Good
73 Fair **74** Poor

Cellular Automata on the IIGS using TML Pascal

Enrico Colombini

Tom Leonard seems to have won the race for the first useful programming environment for the Apple IIGS. His TML Pascal is here, and it works fine. I cannot say the same for the long-awaited Apple-Megamax C compiler (APWC), which lags months behind schedule.

Although I prefer C to Pascal as a development language, I must say that TML Pascal is a very good Pascal implementation, useful for both the casual user and the professional developer. The former has the option to use a simple "console" which makes it easy to write short, simple "Applesoft-style" programs, without worrying about Tools, initialization, and other low-level tasks. The latter gains full access to the power of the Apple IIGS, and to its Mac-like interface.

In this article I'll use the console option, to keep the code short and to be able to concentrate on the essential points. I won't use the Mac-like interface, because I only want to show how to write a simple program on the Apple IIGS. Buttons and menus could easily be added later, enhancing the user interface.

Cellular Automata Revisited

A recent article in *BYTE* magazine (Dec. 86), described the theory of one-dimensional cellular automata, linear relatives of the well-known mathematical game of Life (I'll use the term "linear" instead of "one-dimensional"). That article included a (very inefficient) BASIC program to display cellular automata on the IBM PC's screen.

I made a slight modification to the theory, and wrote a completely different program in TML Pascal to display linear cellular automata on the color screen of the Apple IIGS. This algorithm is much more efficient, and close to the speed of an assembly language program. The main speed limiting factor is, as we shall see, the poor performance of QuickDraw graphic primitives.

A linear cellular automaton is a one-dimensional array of "cells", as shown in Figure 1. Each cell has a "state", identi-

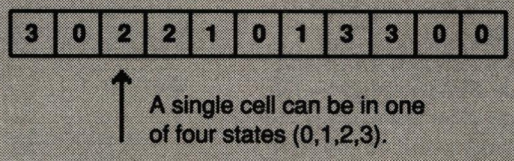


FIGURE 1: A linear cellular automaton is composed by a number of side-by-side cells.

fied by a number associated with the cell. Our cells can be in one of four possible states: 0, 1, 2 or 3.

To make states immediately visible, a different color is associated with each state (Figure 2). For example, a cell in state 0 is displayed as a little white square, while a cell in state 2 is displayed as a red square (no Russian connection).

The array of cells is in theory unlimited, but we'll have to put some restriction on the maximum number of cells, if we want to display it on a real computer screen.

Generations and Birth Rule

Every tick of an imaginary clock, the automaton changes: a new generation is born and replaces the previous one.

The new state of a given cell comes from the old state of the cell itself, and from the old state of its two neighbors (left and right). More precisely, the new state of the cell comes from the sum of the states of these three cells (Figure 3).

Since each of the three cells can have a state from 0 to 3, the sum of these three states is a number from 0 to 9. The state of the new-born cell is derived from this sum, by looking at a table called cellular birth rule (Figure 4). The table gives the new cell state for every possible sum, and is used as shown in Figure 5.

By applying the birth rule to every three-cell group in the original generation, every cell gets its new state. The old generation is then replaced by the new one. The entire process is repeated at every tick of the clock.

Why are linear cellular automata so interesting? Even by representing every state with a different color, it wouldn't be very attractive to look at a changing row of little squares. But if we display every new generation under the previous one, we can see the "history" of the automaton through many generations. In other words, we represent a one-dimensional automaton on a two-dimensional display, where the second dimension (vertical) is the Time. The resulting patterns are quite surprising — they are much more complex than one could expect from the simple mechanism just described. Gliders, pillars, oscillators, quasi-random behavior: every rule gives rise to a different, complex, fascinating pattern of colors.

No Borders

As mentioned, the automaton should have infinite available space in both directions. This is, of course, impossible as we have a finite number of pixels on the screen, and we need at least one pixel to display one cell. We therefore have a problem: what happens at the borders?

The author of the *BYTE* article solved the problem by building a "wall" at each side of the automaton: a cell whose state is always 0. After some experimenting, I found this solution unsatisfactory, because it interferes too much with the "natural" behavior of the automaton. I saw two better solutions: either to extend the automaton far outside the left and right borders of the visible area, or to connect the borders together to create a "circular automaton". I chose this latter solution, because it doesn't create discontinuities of any kind: a sideways moving object (like a glider) is never lost, but reappears from the other side of the viewing area.

You can see in Figure 6 how this affects the birth mechanism: the left neighbor of the first cell is considered to be the last cell and, conversely, the right neighbor of the last cell is considered to be the first cell. Having connected the borders together, there are no more "first" and "last" cells: all cells have equal rights and duties.

State	Color
0	White
1	Green
2	Red
3	Blue

FIGURE 2: A color is associated to each cell state.

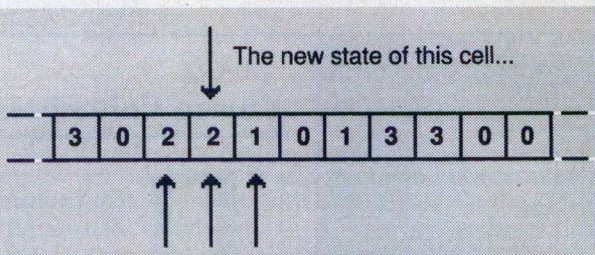


FIGURE 3: The new state of a cell derives from its old state and the old state of its two neighbors.

Sum	0	1	2	3	4	5	6	7	8	9
New State	0	1	2	1	0	3	2	2	2	3

FIGURE 4: An example of cellular birth rule: 0121032223.

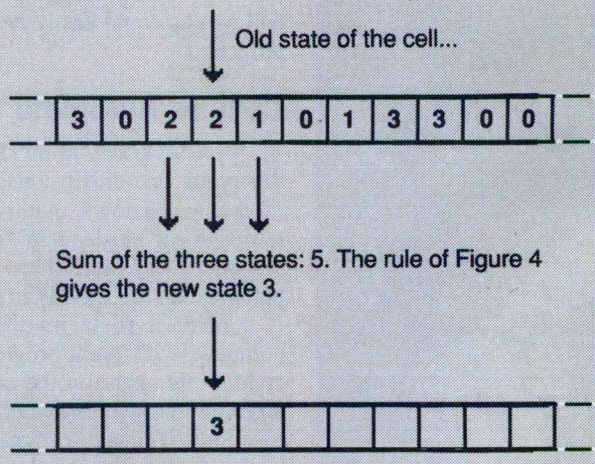


FIGURE 5: How to apply the cellular birth rule of Figure 4 to the automaton of Figure 1.

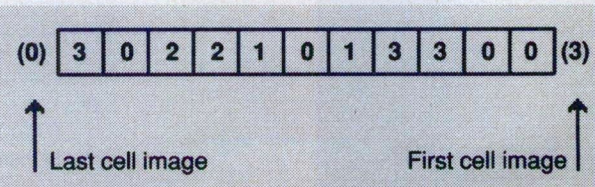


FIGURE 6: The automaton is made circular by considering the last cell adjacent to the first cell.

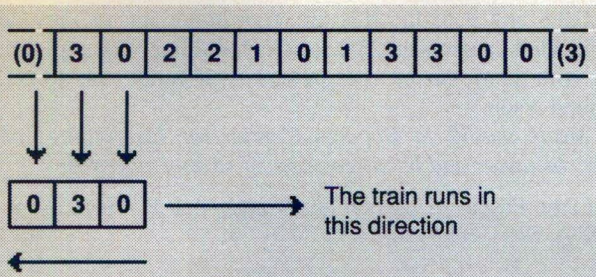


FIGURE 7: The shifting machine algorithm used to quickly compute the next generation.

The Main Program

The name of the program, "Automa", is the Italian word for "automaton".

Thanks to the Console option, the program is rather short. Without it, I should have to initialize the Tools, create a window and handle all the possible events using Toolbox routines. This wouldn't be very complex, but it would hide the interesting points under a lot of "housekeeping" code. Of course, this should be done to create a commercial, Mac-style program. For now, let's stick to the simpler approach.

Note, at the beginning of the program, the compiler directive USES, states the interface and library files to be included. Console IO unforeseeable allows us to access the simple UCSD-style console interface, QDIntf contains the interfaces to the graphics primitives of the IIGS Toolbox, and MiscTools is needed to use the GetTick routine.

The Toolbox is the IIGS' collection of useful routines, grouped by type. For example, all the graphics routines are grouped in what is called the QuickDraw Tool.

The size of the automata (number of cells) and the number of generations are defined by the two constants, Cells and Rows. Because of Pascal conventions, the Main program is the last procedure in the source file. It begins by activating the Miscellaneous Tools (MTStartUp), then uses them to start the random number generator with an unforeseeable number (the state of the Tick counter).

Next, the SetColorTab procedure is called to change the default color table for the 640 pixels mode (the console works in this mode). The original table is designed to allow dithering (a trick to have 16 colors in 640 pixels mode), while we need the same four colors available at every pixel position. The new table is built with the colors in opposite order compared with Figure 2, because the background color (color 0) for the window is blue, while the background color for the automata (state 0) is white. The colors will be reversed in the cell drawing procedure.

The SetUpWindow procedure draws a framing rectangle for the automata, then writes a title and a menu (in the old meaning of the word). In a more refined program (without the console option), this procedure would create a new window, some (true) menus, and the needed controls (buttons).

The RandomRule draws a new random rule. A rule, as we have seen before, is a string of ten digits from 0 to 3, as in Figure 4. The first digit is not drawn at random, but is always zero. This ensures the auto-regeneration of a white (state 0) background (think about it: 0+0+0 = 0, that is: under a white line the new state is the first digit of the rule).

Then, the main program calls the ShowRule procedure, which simply writes the current rule as a sequence of ten digits. After that, the RandomGen procedure draws a new random generation of cells: every cell may have one of the four states with equal probability.

The termination flag "done" is initialized to FALSE, the row counter "row" is positioned on the first line, and the main cycle is entered.

The Main Cycle

The main cycle, which is terminated by setting the "done" flag to the boolean value TRUE, calls the ShowGen procedure to display the current generation of the automaton. This procedure inverts the colors before drawing (by subtracting from 3), to use white (color 3) as background (state 0). Every cell is displayed as a rectangle two pixels wide and a single pixel tall (the best approximation of a square with the IIGS in 640 pixels mode), using a 2 x 1 pixels pen. At first, I used two QuickDraw instructions: Line(0,0) to plot the cell, then Move(2,0) to position the pen for the next cell. Then I noted that the program runs faster by using only the Line instruction, even if it draws a greater number of pixels. So I adjusted it with a test (if...then...else) to avoid plotting over the right border with the last cell of the row.

After displaying a generation, the main cycle passes to the next row, returning to the top of the window when it reaches the bottom. If a key is pressed on the IIGS keyboard, the ExecKey procedure takes care of it. A digit (0..9) changes the rule at the corresponding position (0=first digit or rule for sum 0, 9=last digit or rule for sum 9), 'N' draws a new random rule, 'R' regenerates a "dead" automaton with a random state for its cells (without touching the rule), and 'Q' terminates the program by setting the "done" flag.

The last procedure of the main cycle, NextGen, is the very core of the program: it computes the next generation from the current one. The algorithm is very fast, as it computes the new generation "in place", and minimizes the number of accesses to array elements (a relatively slow operation). It uses a three-cell "train" to remember the last three states, as shown in Figure 7.

Limits of the Program

I admit, the user interface is horrible, more suited to a dull IBM PC than to a graceful Apple IIGS. It would be much better to replace the menu with some buttons, and to change the rule by simply clicking over the digits. I'll leave that to you as a useful exercise.

There is no facility to start with a given pattern. It should be an interesting extension for mathematical users who want to start from a known situation. Although usable, the program is slow; it takes about 100 seconds to draw 172 generations. This comes, alas, from QuickDraw itself. Despite its name, this graphics Tool is very, very slow. Much slower than its Macintosh counterpart, even taking into account the different horsepower of the 65816 compared with the 68000. QuickDraw is very powerful, being able to deal with regions of virtually every shape and dimension, but we must pay a lot for that power. Unless Apple will rewrite it to run at least five times faster, I think that nearly all developers will be forced to bypass it, writing directly into the display memory. Especially game developers, like myself.

How much faster could this program run by directly accessing the display memory instead of using QuickDraw routines? Without doing any calculations, I'd say... about 50 times faster! That is, 2 seconds from top to bottom of the

window (172 generations). Oh, I forgot, without using any assembly language instruction (but using C: I don't know the low-level efficiency of Pascal, though TML seems quite good).

Playing with Automata

Apart from its mathematical interest, the program is funny to play with: every rule seems to produce a different behavior. After some time, you'll begin to note that the patterns generated by the automata tend to be variations of a few basic types. I'll give you some rules to illustrate the most common types:

- Color over all (0220202100): a single color (included white) destroys all others in a few generations, giving rise to a rather boring picture. A new random generation ('R') has a small probability of survival. After this situation, when you change rules you must also give an 'R' command to restart the dead colors.
- Alternate colors (0020333202): the background changes between two or more colors, with or without other types of patterns.
- Pillars (0000032232): only stable structures survive, leaving vertical bars, sometimes of a single color, sometimes of two or more.
- Triangular chaos (a common situation): a lot of upside-down triangles, which never seems to repeat the same pattern. Often seen in two colors (0020000122), it is not rare with three or four of them (0203212033).
- Multicolor chaos (0221033100): sometimes it produces a nice Persian carpet.
- Chinese decorations (0310210230): a variant of multi color chaos, in which beautiful regular structures appear from the apparent chaos at irregular intervals. Some varieties don't survive very long.
- Oscillators (0202110311): often associated with pillars, these structures repeat themselves at regular intervals. The period of the oscillators (number of generations before repetition) is typically from 2 to 12, but we have discovered oscillators with a period of about 250 generations! Try 0010232133 to see tropical oscillators.
- Gliders (0020303110): gliders are moving oscillators, that travel in a given direction. When they encounter an obstacle (like a pillar), they can destroy it, turn back or simply die. They are not uncommon, but are sometimes masked by other elements. They range from very simple to very complex structures, the later often called spaceships (another term coming from Life). The given rule has a good chance of creating a huge spaceship, bouncing between two pillars, apart from many short-lived gliders.
- Slow stabilization (3001021332): after much time of constant changing, stable structures (pillars and oscillators) will dominate the field.
- Struggle for survival (0110312212): doomed configurations with little hope to survive. Sooner or later, despite their fierce fighting, the evil triangles will eat them.

There are also many strange automata, resulting from a combination of the basic types, or outside any type. Some examples: 2020213131 (earthquake), 1010123232 (baobab branches), 0121032223 (Christmas wallpaper), 0201123302 (hard to define). Try them!

You could explore the endless automata universe using chance (the 'N' command) or reasoning. For instance, the digits 0,3,6 and 9 of the rule are very important, as they determine what happens under a solid colored line. Try experimenting with them, but be very patient — it is not easy to get the desired result.

Many thanks to Roberto Cerruti for his findings about rules behavior and his discoveries of strange patterns.

(Tml Pascal is available through A.P.P.L.E. Co-op and APDA. See page 5)

//

Please rate this article on the Reader Service Card by circling:

81 Excellent **82** Good **83** Fair **84** Poor

```
(*
   Automa: linear (1-dimensional) cellular automata

        (c) Dinosoft 1987 by Enrico Colombini

This program can be freely distributed if copyright note is not altered
Cellular automata theory is described in Byte magazine, december 1986.
These automata don't live on a plane, but on a cylindric surface:
right and left edges of the display rectangle are connected.

*)

PROGRAM Automa(input,output);

USES  ConsoleIO,
      QDIntf,
      MiscTools;                    {used for tick counter}

const Cells = 150;                  {number of cells}
      Rows = 172;                   {number of visible rows  (generations)}
      BaseX = 10;                   {upper left cell of first generation}
      BaseY = 5;

var rule:  array[0..9] of integer;  {cellular birth rule}
    gen:  array[0..Cells] of integer;  {current generation, with  "carry"}
    row:  integer;                  {current row}
    done:  boolean;                 {quit flag}

(* Rnd: draw a random integer in the range min..max *)

function Rnd(min,max: integer): integer;

begin
   Rnd := (min + abs(Random) mod (max - min + 1));
end;


(* RandomRule: draw a new random rule; the first number is always 0 *)

procedure RandomRule;

   var i: integer;

begin
   rule[0] := 0;                    {keep white background}
   for i := 1 to 9 do rule[i] := Rnd(0,3);  {draw in range 0..3}
end;


(* ShowRule: show current rule *)

procedure ShowRule;

   var i: integer;

begin
   GotoXY(45,5);
   Write('Current rule:  ');
   for i := 0 to 9 do write(rule[i]);
   ClearEOL;
end;


(* RandomGen: draw a new random cells generation *)

procedure RandomGen;

var i: integer;

begin
   for i := 0 to Cells-1 do
       gen[i] := Rnd(0,3);:         {random color distribution}
end;


(* ShowGen: draw current cells generation at row <row> *)
```

```
procedure ShowGen(row: integer);

var i: integer;

begin
    MoveTo(BaseX,BaseY+row-1);
    SetPenSize(2,1);                    {double pixel for mode 640}
    for i := 0 to Cells-1 do begin
        SetSolidPenPat(3-gen[i]);       {set color (0=white)}
        if i<Cells-1 then
            Line(2,0)                   {draw cell and move}
        else
            Line(0,0);                  {draw last cell}
    end
end;


(* NextGen: compute new generation of cells. It does the whole work! *)

procedure NextGen;

    var i: integer;
        a,b,c: integer;

begin
    a := gen[Cells-1];                  {start shifting machine}
    b := gen[0];
    gen[Cells] := b;
    for i := 0 to Cells-1 do begin      {for every cell}
        c := gen[i+1];                  {read right cell value}
        gen[i] := rule[a+b+c];          {compute new cell value}
        a := b;                         {shift original cells}
        b := c;
    end
end;


(* SetUpWindow: draw automata framing rectangle and user menu *)

procedure SetUpWindow;

    var r: Rect;

begin
    SetSolidPenPat(0);                  {framing rectangle}
    SetPenSize(1,1);
    SetRect(r,BaseX-1,BaseY-1,BaseX+Cells*2+1,BaseY+Rows+1);
    FrameRect(r);
    GotoXY(45,2);                       {title & user menu}
    Write('\/\/ Linear Automata \/\/');
    GotoXY(45,3);
    Write('(c) 1987 by Enrico Colombini');
    GotoXY(45,7);
    Write('0..9)        Change rule');
    GotoXY(45,8);
    Write('N)           New random rule');
    GotoXY(45,9);
    Write('R)           Random Generation');
    GotoXY(45,10);
    Write('Q)           Quit');
end;

(* SetColorTab: setup color tab for 4 plain colors *)

procedure SetColorTab;

var tab: ColorTable;
    c: ColorValue;
    i: integer;

begin
    tab[0] := $000f;                    {blue = background}
    tab[1] := $0f00;                    {red}
    tab[2] := $00f0;                    {green}}
    tab[3] := $0fff;                    {white}
    for i := 0 to 3 do begin            {repeat four times}
        c := tab[i];
        tab[i+4] := c;
        tab[i+8] := c;
        tab[i+12] := c;
    end;
    SetColorTable(0,tab);               {change active table}
end;


(* ExecKey: get key command and exec it *)

procedure ExecKey;

var c: char;
    d: integer;

begin
    c := ReadChar;;
    if c in ['0'..'9'] then begin               {change rule}
        d := ord(c) - ord('0');                 {get rule digit position}
        inc(rule[d]);                           {increment rule digit}
        if rule[d]>3 then rule[d] := 0;         {cycle in range 0..3}
        ShowRule                                {show new rule}
    end else if (c='N') or (c='n') then begin   {new random rule}
        RandomRule;
        ShowRule
    end else if (c='R') or (c='r') then         {random generation}
        RandomGen
    else if (c='Q') or (c='q') then             {quit}
        done := true;
end;


(* Main program *)

begin
        MTStartUp;                      {init MiscTools for clock}
        SetRandSeed(GetTick);           {init random generator}
        SetColorTab;                    {change colors}
        SetUpWindow;                    {prepare window & menu}
        RandomRule;                     {initial rule}
        ShowRule;
        RandomGen;                      {initial generation}
        done := false;                  {clear quit flag}
        row := 1;                       {start with first row}
        repeat                          {begin main cycle}
        ShowGen(row);                   {show current generation}
        inc(row);                       {position for the next}
        if row>Rows then row := 1;      {restart if last row}
        if keypressed then ExecKey;     {if command, exec it}
        NextGen;                        {compute next generation}
    until done;                         {repeat until quit flag}
    MTShutDown;                         {close MiscTools}
end.
```


MODEM MGR® **$50.00**

plus $3 ship/hand • plus tax (Calif. only)

- For Apple][+, //c, //e, IIGS
- Both DOS 3.3 and ProDOS support
- Supports most serial and modem cards
- Supports most smart and non-smart modems
- Fast operation (110 to 19200 baud)
- Text capture and auto disk-save
- Screen capture (snapshot)
- Full-screen editor
- Full- or split-screen display with block send (for CB, chat, packet radio)
- Macro script files for auto log-on
- Programmable function keys
- XMODEM and BINARY II file transfer
- Supports Apple or Videx 80-column card
- On-screen time-of-day or elapsed-time display (with most clock cards)
- Unattended mode with password protection
- Emulates DEC VT-52, DG 200, others
- Supports most printer cards
- Choice of carrier or "smart" modem control
- Not copy-protected
- Available direct only. Send check to:

MGR SOFTWARE

Suite 101
305 So. State College Blvd.
Anaheim, CA 92806
(714) 993-0294

Apple, ProDOS are trademarks of Apple Computer Inc.

CIRCLE **9** ON READER SERVICE CARD

Comparison of Three Cad Programs:Part 2

Lee Webster

In Part 1 we began our look at three CAD programs for the Apple II by discussing the details of *CADDRAW* and *discoverCAD*. We will conclude our discussion with a look at the details of *MATC-CAD*, along with a comparison of the documentation of all three programs. Finally, we'll provide a comparison chart of the CAD features of the programs, and give our overall conclusions.

MATC-CAD

The *MATC-CAD* package was developed at the Milwaukee Area Technical College (MATC) to simulate the general two-dimensional drafting features of Computervision's CADDS3 system. In a way, then, here is an Apple II simulation of a minicomputer CAD system! It is relatively powerful, but you pay for it, as it is harder to learn to use.

Operating Environment

A variety of input and output devices are supported (*see Table 1 from March 1987 Call -A.P.P.L.E. pg. 11*). Even with a minimal configuration (mouse and dot-matrix printer) you can still use this program — at a much lower hardware cost than the Computervision system it simulates!

When you boot the program (program disk in drive 1, data disk in drive 2), you get a title screen and licensing information (warning?!). After selecting the input device and entering another command, you are in the main program where you can choose to start a new part (drawing) or work on an old one. You now have the standard MATC-CAD prompt, the # symbol.

Command Structure

This CAD package is command driven, which takes a little longer to learn (vs. menu driven), but allows for plenty of flexibility and power. The syntax of a command is essentially:

Verb-Noun (Modifier) Phrase : Data Input Section

The part to the left of the colon includes a verb (such as Insert, Trim, Move, etc.), a noun (such as Line, Arc, Circle, Point, etc.), and then an optional modifier which further restricts or specifies a command (such as Vertical, Tangent to, Angle, etc.). To the right of the colon you enter location and/or entity (i.e. any line, arc, circle, etc.) information. The location data may be entered in three ways:

a. by digitizing with the graphics tablet, joystick or mouse,
b. entering cartesian or polar coordinates (absolute or relative), or
c. coordinates based on the position of an existing entity.

The commands are either typed in from the keyboard, pointed to with the tablet pen on a graphics tablet menu, or pointed to on the screen menu with the mouse. I have worked with the Apple Graphics Tablet and it performs well in this mode. It is especially good for those who don't type! You even have the option of creating your own custom tablet menu, if the one supplied doesn't suit you.

The screen menu for the mouse is displayed at the bottom of the screen. Initially, a list of verb abbreviations appears. With the mouse you can highlight one and press the button to select. The command is written to the prompt line on the screen and the next menu (i.e. nouns) appears. At each level, applicable commands appear on the menu including digits, punctuation, return key, etc. It works well, again good for non-typists. It is slow in entering several digits for a value, so you can type them on the keyboard instead. In fact, you can combine keyboard entry and mouse selection in a single command line. Table 1 contains the command summary.

When locating points or referring to entities, you *must* use the tablet pen, joystick, or mouse to point — the keyboard arrow keys don't work. The only exception is if you want to input absolute or relative coordinates to locate points, in which case you must type the values in.

The screen operates in three modes. Initially it shows the drawing area with 4 text lines at the bottom (or screen menu if using the mouse) where the last three commands you typed are shown and a line is provided for you enter your next command. You can toggle to a full drawing screen (no text lines) or full text screen (shows a screen full of your last 24 commands — this can be helpful in referring back to a previous action so you can repeat it). Whether typing or using the tablet, if you enter the wrong characters or wrong sequence (like noun before verb) the computer beeps, you hit Return and try again! The software runs reasonably fast, although there are the frequent disk accesses common to Apple Pascal. (It is possible to install MATC-CAD on a Corvus hard disk system.)

Drawing Common Entries

Let's look at some of the commands available for drawing common entities:

Lines: Lines can be drawn from one location to another, from the end of one existing line to the end of another, at a specified length, or horizontal or vertical. For example, the command for a vertical line is:

#INS LIN VER: DIG d1 d2

where # is the prompt (computer supplied), INS is the verb Insert, LIN the noun for Line, VER the modifier for Vertical. DIG is computer generated, d1 means that you point (with tablet pen, joystick, or mouse) to the first point (where you want the line to start) and d2 means you point near where you want the line to end. The program will automatically make it vertical stopping opposite your d2 point. There are special commands for drawing rectangles and squares; lines parallel or perpendicular to other lines; and lines tangent to circles. Two line fonts are available: solid and dashed (for hidden lines in drafting). It would be useful if there were a centerline font too, but the manual suggests doing centerlines as a series of line segments — possible yes, but not that convenient!

Circles: Circles can be defined in a number of ways, including: 3 points, radius, diameter, or tangent to an existing entity (Figure 1 part A). Depending on the method, you need to specify data by keyboard or digitize positions.

Arcs: Arcs are drawn in methods similar to circles plus there is the ability to draw semi-circles directly and to specify beginning and ending angles for an arc.

The one problem with these entities is that you don't see the result until it is drawn (no rubber-banding or previewing). So, if it is wrong, what do you do? Two options: erase it (by digitizing on the entity(ies) you want erased (or putting a window around a large area you want erased), or by editing the line. Editing means that you can relocate one or both ends of a line, trim a line to the intersection of another entity, or divide a line into segments (so you can manipulate them separately). Similar editing is possible for circles and arcs, including the ability to take an arc and turn it into a circle! These all work well, except that to erase a rectangle (drawn with the rectangle command), you have to digitize all 4 sides — it would be better to have a single command to do this. There is also the helpful Unerase command, should you change your mind after erasing something!

TABLE 1: MATC-CAD Command Summary

ENTITY CREATION:
Insert horizontal line
Insert vertical line
Insert perpendicular line
Insert parallel lines
Insert tangent lines
Insert lines of given length
Insert line at givenangle
Insert rectangle
Select line font (solid, dash)
Insert circle (radius, diameter, arc expanded to circle)
Insert circle tangent to line, arc, circle
Insert arc (radius, diameter, start & end angle)
Insert point (also at intersection of lines)
Insert fillet (radius, with or without trim)
Insert splines (tangent to starting or ending line, closed loop)
Coordinates (absolute, relative Cartesian, polar)

SCREEN CONTROL:
Zoom (up, down, all, window)
Page[Pan] (display, up, down, left, right)
Save display
Restore display
Set grid
Show grid
Snap to grid
Select layer
Echo (make visible) layer
List layer (show number and status)
Insert parts (scale, rotation, layer)
Insert figures (grouped entities)

PLOTTING/PRINTING:
Select plot
Echo frame
Plot (HP: 7470A, 7475A)
(HI: DMP40, DMP41, DMP51)
(Apple Color Plotter)
(Epson or Panasonic printer)

TEXT & DIMENSIONING:
Insert linear dimension (horizontal, vertical, point to point)
Insert radial dimension
Insert diameter dimension
Select dimension (precision, arrow placement, extension lines suppressed)
Change dimensions
Insert text (height, width, slant, angle, left-center-right justified)
Select text
Change text

EDITING:
Trim (line, circle, arc) to intersection
Trim (line, circle, arc) to extend length of entity
Trim (line, circle) into two or more parts
Measure (distance, angle, area)
Verify (show data on entities)
List parameters
Mirror
Mirror and copy
Move
Move and copy
Rotate (copy, angle)
Erase (all, line, circle, point, entity)

Adding Text

Text is inserted quite easily with a single command. There are many optional modifiers, which give you additional control over the appearance of the text. For instance, you can specify the height, width, slant of letters and whether the text is to be left-, right-, or center-justified. You can also set the angle of the text so it doesn't always have to be horizontal (Figure 1 part B). Labels with arrow leaders (for putting

notes on drawing parts) are also possible. There are commands in the manual for changing text parameters, but not for the string of the text — an important omission.

Special Lines

When doing engineering drawings, you often need to draw fillets (rounded corners). MATC-CAD has that capability, using a default radius for the arc or letting you specify the radius. The tangent lines will be automatically removed, unless you specify that you want them left (Figure 1 part C). You can even have fillets drawn between lines and arcs. In Figure 1 part D there is a straight line and an arc. The fillet was automatically drawn from the line to the arc.

Another common drawing technique is a spline. This is a smooth continuous curve that passes through a series of 3 or more defined points. These are easily done with MATC-CAD.

Dimensioning

With MATC-CAD, the measurements or dimensions are calculated automatically. The user only need specify the form of the dimension note and its position. Horizontal and vertical dimension lines are possible (with the extension lines and arrowheads added automatically). You simply point to the two entities you want to dimension between and indicate where you want the dimension line (the line containing the dimension value or number) to go, and "zap", there it is! The arrow heads can be located either inside the extension lines (Figure 2 part A) or outside (Figure 2 part B). You can also specify the degree of precision for the measurements and add other text. The height, width, angle, etc. of the lettering may be specified too.

Circles may be dimensioned by their radius (Figure 2 part C) or their diameter (Figure 2 part D). Dimensioning of angles is also possible.

Screen Control

With MATC-CAD, as with most CAD systems, you can control the screen display size and position. For example, you can Zoom in or out, or shift the screen window to look at another area of the drawing, etc. These actions take a few seconds to accomplish as the computer must redraw the entire screen. You can also set up a grid to help you draw — particularly useful when working with a joystick for input. The grid spacing is user selectable (you can even have a different spacing horizontally from what you have vertically). The grid can be visible (unfortunately a 2-step process since it needs a screen redraw), or invisible, and you may select the origin of the grid. Finally, you can elect to have the Snap feature **On** or **Off.** When Snap is **On,** the program uses the closest grid location to the one you digitize.

FIGURE 1

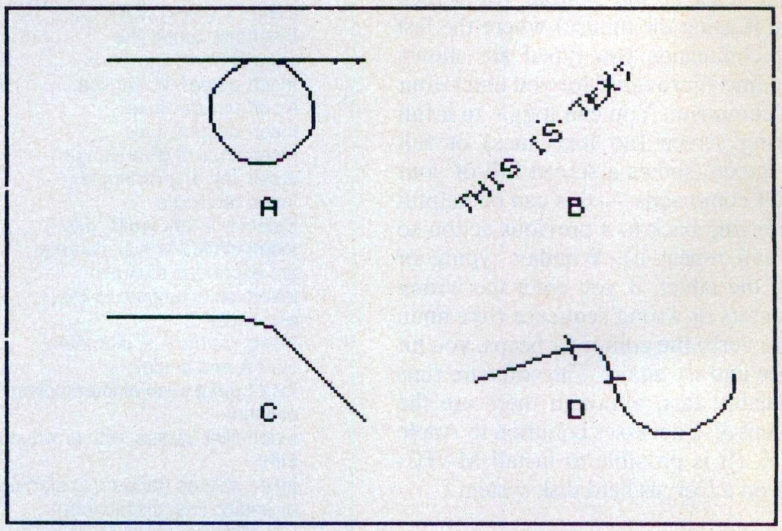
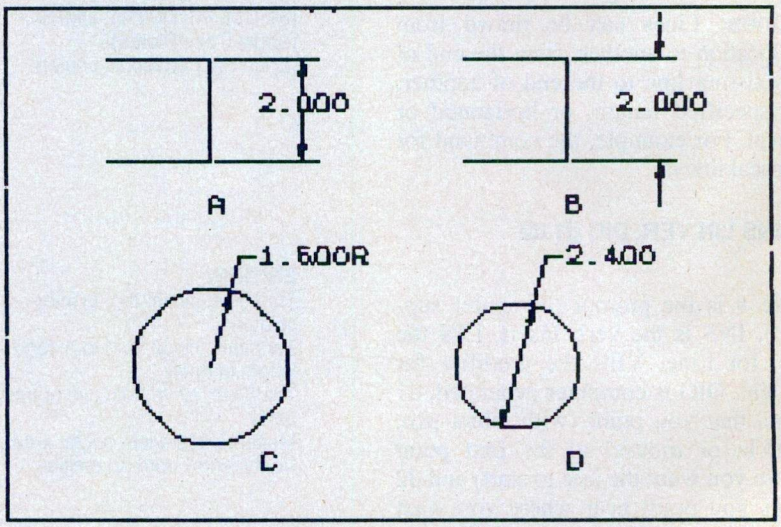


FIGURE 2



Other Powerful Commands

Layering: With layering, you can work on any of 256 layers. One, all, or a combination can be displayed at one time. It is very convenient, for example, to do the object lines on one layer, dimensioning on another, and borders/title blocks on another.

Parts & Figures: When drawing circuit diagrams and similar items, you use a number of entities over and over (like resistor symbols, transistor symbols, etc.). By saving subdrawings to disk, you can have them available later to insert into an existing drawing.

Mirror, Move & Rotate: With a symmetrical object, you can draw one half and use the Mirror Command to generate the other identical side. An entity (or window with several entities) can be moved to another location in the drawing quite easily. If instead of moving, you want to copy it, that is also

possible with a modifier on the Move Command. An equally useful feature is the Rotate Command which can be used, for example, to place a number of equally spaced circles along the circumference of a defined circle (such as a ring of bolt holes).

Measure: By specifying the Measure Command and identifying the entities, you can get the angle between 2 lines, angle from a line to the horizontal, the distance between parallel lines (if you goof and ask for the distance between two lines that intersect, the program tells you!), or even the area bounded by a series of lines and arcs.

Help

If you want details on a particular entity, you can identify it (by command and digitizing to select it) and a text display will show the line type, X and Y coordinates of end, and length. For arcs and circles, the information given includes layer, font, X and Y coordinate of origin, radius, and beginning and ending angle.

For an overall display of the current parameters (layer, line font, number of entities, coordinate of lower-left corner, screen scale, text parameters) there is a separate command. All of the above are helpful in seeing where you are as you plan additions or changes to a drawing.

Printing

A draft copy of the drawing can be created by a supported dot-matrix printer (*See Table 1 March 1987 Call - A.P.P.L.E*). Print drivers are included in the software — you need only choose the proper printer from the list presented on-screen. The drawing is only 3" x 5". You might be able to get a larger printout *if* you know a way of trapping the screen image for use with a separate screen dump program.

Plotting

MATC-CAD supports several plotters (*See Table 1, March 1987 Call - A.P.P.L.E. pg. 11*). Using the Houston Instruments DMP41 or 51 it is possible (according to the manual!) to get an active plotting area of 20" x 32". Figure 10 was plotted on an HP 7470A plotter. On HP plotters you can also select the pen number for plotting to obtain multicolor plots.

Utilities

What would we do without utilities? In MATC-CAD, they reside outside the CAD program proper (so you must first exit, saving your drawing). You then are able to format a disk (remember they are Pascal data disks!), pack files to save space, name a disk, transfer and rename files, list files, change the date, and copy a disk. (Although this option is not shown on the menu, it is mentioned in the manual.)

Manuals

Each program's approach to documentation is as varied as the programs themselves. A summary of the manuals for each program follows:

CADdraw

Two manuals come with the package. One is a 5 1/2" x 8 1/2" stapled 10-page discussion of "Helpful Hints on Computers and Kitchen Sink Software". It covers basic definitions, getting started, making back-ups, initializing a disk, etc. for the novice. Also included is information on legalities, licensing, updates & replacements, and other Kitchen Sink Software policies.

The other manual, same size, 60 pages long, covers the details of the program. It begins with an adequate mini-tutorial to lead you through the basics of the program. Next, there is a section describing details of the three modes, bottom-of-screen displays, and commands. The information is there, but not in a particularly organized manner. This could be improved in a more "Reference Manual" approach. Also missing is an index to topics.

discoverCAD

The discoverCAD manual is comb-bound, with 55 8 1/2" x 11" pages. It may have been done with a Macintosh and LaserWriter, as it is attractively presented, easy to read, with boldface highlighting and example diagrams. The instructions are simple and direct although there is considerable repetition on pointing, clicking, dragging, etc. with the mouse to select commands. Although it does have a good table of contents, there is no index. The first chapter gives a good tutorial on basic principles. Each of the remaining chapters essentially covers each of the commands on the Menu Bar. The appendix has the necessary details for setting the Apple Super Serial Card, installing the plotters (i.e. what the DIP switch settings should be for each), and a diagram of the slot configurations for the Apple IIe.

MATC-CAD

Because MATC-CAD was developed as a teaching tool, there is a "Student Manual" and an "Instructor Manual". Both are 8 1/2" x 11" comb-bound, soft-cover typed manuals.

The "Student Manual" has about 230 pages. After chapters on the Fundamentals of Computer Graphics and the Basics of the MATC-CAD Syntax, there are 12 sections or teaching units. Each contains an introduction, objectives, a table of contents, a reference section (describing the commands), an exercise section (normally fill-in-the blank questions that don't need to be done using the computer), and a lab exercise section (drawings or activities to do with the computer). The order and organization of these units are well planned, introducing some basic commands in the earlier units and returning in a later unit to talk about the additional "bells and whistles" available with those commands. There is a minimum of verbiage in the manual — instead just brief statements, example command lines, and labeled diagrams. It works well as a learning manual — hands on, with feedback through the exercises. Systematically done! There is even a brief section outlining the differences between CADDS3 (which this package simulates) and Computervision's CADDS4 package. What it is not, is a reference manual. There is no index, so you must look in the table of contents to locate topics. MATC has recently produced, however, an 8 1/2" x 11" cardstock (2 sided) Quick Reference Card which lists most of the commands and their syntax — it even includes the small diagrams explaining what needs to be digitized for the commands. This is a very helpful addition — it could be improved only by adding Student Manual page number references for when you want additional information. I found, after getting familiar with the package, that I could do a lot of work just referring to the Quick Reference Card with only occasional trips to the manual.

The "Instructor Manual" has about 165 pages. It covers the hardware (setting up, aligning the graphics tablet), software (disk backup, error messages), and teaching helps. The latter includes a sample course syllabus, unit notes (explaining what to emphasize and where problems are likely to occur in each unit), and sample lab exercises and final projects (additional ones that can be assigned that are not in the Student Manual). The answers to the

exercise sheets in the Student Manual and sample quizzes and exams (with answers!) are also included. In the back there is even a handy section on testing and repairing the Apple Graphics Tablet Pen which has a tendency to break.

The package cost also includes a year of software maintenance and support. On the few occasions I have needed to call MATC, their personnel have been most helpful. Updates (included in the maintenance agreement) seem to occur every 12-18 months. (I first used this program when it came on 4 disks — they have it all down on 1 disk even though there are more commands.)

Conclusion

CADdraw

CADdraw has a lot of features for a low-cost CAD package. It can also be run on any Apple II model without any additional equipment (except a printer, which most folks already have). It can be configured to use several other input devices (the mouse is the easiest to use). You can do some decent drafting with this program, but are limited to a dot-matrix printout of the results. If there were a way to drive a plotter (or laser printer), you might use this as a productive CAD package. However, it has limited scales available, and certainly none of the "power" features found in professional CAD packages. It is a good low-cost introduction to CAD and would have limited use in real-life drafting situations, as it lacks plotting and automatic dimensioning.

CADdraw is a good choice for an introduction to CAD in an educational setting, as many institutions have Apple II's available. The cost is low, and best of all, it has a *superb* site licensing arrangement — $5 per copy! New users should be able to learn the program easily after only a brief introduction. The command summary sheets are handy (there are several in the manual).

discoverCAD

This program is easy to learn and to use, so it would work very well in an educational setting. It has good plotter support (making it a possibility for "real world" use) and documentation. The similarities to the Macintosh operating environment are of particular interest. Commands are logically arranged, and such features as Masks are very useful in getting precise drawings.

On the negative side, there is annoying disk shuffling when loading or saving drawings. The operating speed is

TABLE 2: Comparison of Major CAD Features

Feature	CADDRAW	MATC-CAD	discoverCAD
ENTITY DRAWING:			
Lines (horiz, vertical)	Yes	Yes	Yes
Parallel lines	No	Yes	Yes
Perpendicular lines	No	Yes	Yes
Lines at given angle	No	Yes	Yes
Lines tangent to entities	No	Yes	Limited
Lines to/at intersections	No	Yes	Limited
Circles	Yes	Yes	Yes
Arcs	Yes	Yes	Yes
Circles/Arcs tangent to other entities	No	Yes	Limited
Fillets	No	Yes	No
Sketch mode	Yes	Yes	No
Irregular curves (splines)	No	Yes	No
Hatching (section lines)	w/symbol tbl	No	No
Line types	2 & usr defnd	solid, dash	solid, hidden
Line weights (widths)	9	1	1
Rectangles	Yes	Yes	No
Ellipses	Yes	No	No
Shapes--predefined	500 included	No	No
Shapes--user defined	Yes	No	No
Isometric mode	Yes	No	No
EDITING:			
Trim line/arc/circle	No	Yes	Yes
Erase entity	Yes	Yes	Yes
Move entity	Symbols only	Yes	No
Copy	Symbols only	Yes	Yes
Mirror	Yes	Yes	No
Mirror and copy	1/2 screen	Yes	Yes
Rotate	Symbols only	Yes	No
Rotate and copy	Symbols only	Yes	Yes
Measure angle	No	Yes	Yes
Measure distance	Indirectly	Yes	Yes
Measure area	No	Yes	No
TEXT & DIMENSIONING:			
Linear dimensions	No	Yes	Yes
Radial dimensions	No	Yes	Yes
Diameter dimensions	No	Yes	No
Leader dimensions	No	No	No
Tolerancing	No	No	No
Notes	Yes	Yes	Yes
Scaling	Yes	Yes	Limited
Text insertion	Yes	Yes	Yes
Text definition			
upper/lower case	Yes	No	No
height	No	Yes	Yes
width	No	Yes	Yes
slant	No	Yes	Yes
left-justified	No	Yes	Yes
right-justified	No	Yes	Yes
center-justified	No	Yes	Yes
angled text (degrees)	0, 90, 180, 270	0-->360	0-->360
Modify/change text	No	Yes	No
SCREEN CONTROL:			
Define & draw grid	Yes	Yes	Yes
Snap grid	Yes	Yes	Yes
Layers	2	256	128
Zoom	No	Yes	Yes
Pan	Page 1 to 2	Yes	Yes
Redraw	No	Yes	Yes
Rubber-banding	Yes	No	No

adequate, except when using a small grid and then zooming or panning. Diameter dimensioning is missing, as is a pure Move command (without copying). On some linear dimensioning, there is overlap of dimension values and dimension lines which produce clutter on the output *(Figures 6 and 7. See Call - A.P.P.L.E. March 1987).*

MATC-CAD

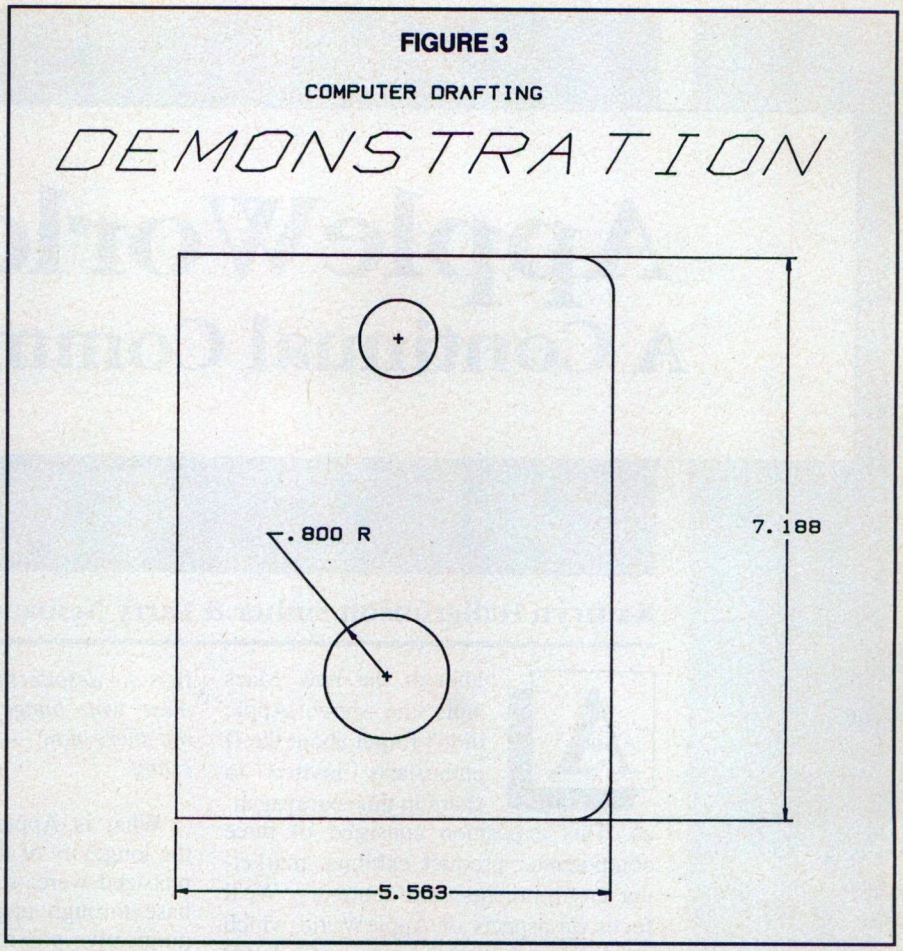
For the market for which it is designed (namely education), this is a very well executed program. The Student Manual carefully and systematically leads the user into the basics of CAD operation. As a bonus, you are exposed to the syntax and operational features of a minicomputer CAD package. Add to this the ready availability of Apple IIs in many schools, colleges and universities, and it means there will be little additional hardware expense. (For those non-loyal Apple folks — there is also an IBM PC version of MATC-CAD!)

On the other hand, I think this package is good enough that those looking for a way to use an Apple II for CAD in a real-world situation might consider it. (In the IBM PC market, there are more contenders as good basic CAD packages.) With plotter output, you can generate first-rate drawings (in multiple colors if you use different layers, different pens and your plotter has good repeatability). It has some limitations: slowness (but not really bad) of the Pascal, crude text editing capability, no easy way to draw centerlines, no section lining (hatching), and the fact that in some cases the extension lines on dimensions touch the object lines (Figure 10) (a "no-no" for those who remember their old drafting instructors!)

And Finally...

CADdraw is best suited for an educational setting because of its low cost (especially for multiple copies), and minimal hardware requirements. discoverCAD is the easiest of the three programs to learn and would be an excellent choice for educational institutions with access to 128K Apple IIe or IIc systems. Because it does have plotter support, its abilities (although limited) may even make it suitable in a work environment. MATC-CAD may be the hardest of the three packages to learn, primarily because of the many commands available. Plotting support, student instructional manual, and similarities to a minicomputer CAD system make it a good choice for advanced educational or training situations. It is the best of these three CAD packages for those who want to use an Apple II system for day-to-day drafting chores.

FIGURE 3



As everyone has different drafting needs, one way for you to decide which (if any) of these Apple II packages would be satisfactory would be to examine Table 1 (hardware requirements) and Table 5 (major features)—published in *Call -A.P.P.L.E. March 1987.* In Table 5, a "yes" indicates that a separate command or straightforward procedure is available.

Frankly, none of these packages match the Macintosh drafting software, but after all, the Mac *is* a graphics machine. It's too soon to tell whether the Apple IIGS will foster new CAD programs to take advantage of its enhanced graphics capabilities, but if so, there could be *new* life for the Apple II family in CAD.

Please rate this article on the Reader Service Card by circling:

91 Excellent **92** Good

93 Fair **94** Poor

Product:
CADDRAW
Company:
Kitchen Sink Software
903 Knebworth Ct.
Westerville, OH 45081
(614) 891-2111
Price: $60
CIRCLE **88** ON READER SERVICE CARD

Product:
discoverCAD
Company:
Hearlihy & Co
714 W. Columbia
P.O. Box 869
Springfield, OH 45501
1-800-622-1000
Price: $209
CIRCLE **89** ON READER SERVICE CARD

Product:
MATC-CAD
Company:
MATC-CAD
1015 N. 6th Street
Milwaukee, WI 53203
(414) 278-6743
Price: $395
CIRCLE **90** ON READER SERVICE CARD

AppleWorld '87
A Continual Commitment

Kathryn Hallgrimson Suther & Larry Neibauer

Although the new Macs stole the show, Apple didn't forget about the II enthusiasts invited to share in this extravaganza. This exposition consisted of three components: product exhibits, marketing environments, and seminars. We'll focus on aspects of AppleWorld which related specifically to the Apple II family.

Of special interest, is the **Apple II Panel Update** transcribed by Larry Neibauer. This discussion was moderated by *Tom Virden* and took place between such well-known Apple employees as: *Harvey Lehtman, Neil Amsden, Rich Williams, Rob Moore, and Mike Smith.*

Features, Benefits, and Positioning

On the first day, Mike Smith held an Apple II seminar to discuss features, benefits and positioning of the Apple II. Mike is in charge of Product Management for the Apple II line, maintaining current and new developments on the Apple IIe computers. He also manages a group of people who work on the Apple IIc and IIGS. He began his discussion with the statement:

"Our charter, as if you didn't know, is really to try to perpetuate this theme of 'Apple II Forever'. *Sometimes I wonder whether this is a reasonable thing to do or not, but to tell you the truth, after talking to a few people here and looking at what people do with the Apple IIe and* IIGS *— despite the fact that there's all these whiz bangey Macintosh products out there now — is really pretty interesting."*

What is Apple's position to ensure the longevity of the II? Two points emphasized were: supporting the installed base through upgrades whenever economically possible, and maintaining compatibility. We all are aware of the efforts Apple has made to allow the GS to run most Apple II software. We've seen both the IIe and IIc continue to evolve: the IIe with the new keyboard, the IIc with the expanded memory. Mike Smith and all the speakers from Apple stood behind the statement expressed by **Jean-Louis Gassée**:

"If you have a product that sells, you don't get rid of it, you keep improving it."

Where does Apple position the IIe, IIc, and IIGS? The IIe has an eminent place in K-12 education. 300,000 IIe's are sold a year, testifying to the appeal of this machine. The IIc is targeted to the home users. It has also been recognized as a seasonal product. Of the 150,000 IIc's purchased a year, 80% are purchased between September and December. At this point in time, Apple isn't sure of the position the IIGS will take. They suspect it will be a 50/50 split between the consumer and education market. Time will tell...

Marketing Environments

One of the most interesting features of this year's AppleWorld was the

'Marketing Environments'. The entire lower level of the Westin Bonaventure Hotel was set up to demonstrate real tasks Apple computers perform. What made this an effective presentation, was the fact that real people — teachers, engineers, children, etc. — came together from different parts of the country to share their specific applications.

At one end of the room we saw an engineering business using Macintoshes and InBox (a product produced by Think Technologies) to communicate information between different divisions of the office. (This business also had offices in Tokyo and Paris which they communicated with using Telex.) Within their home office, if changes needed to be made to a product, the information could be easily passed to the different divisions. They demonstrated how these changes can be made, revisions sent back (via InBox) for approval, and the go-ahead given to begin production. It seemed like a pretty efficient way to manage inter-office communication.

There also were a number of different office type environments set up, a law firm, a doctor's office, higher education administrator's office... but what seemed to draw everyone's attention was the K-12 education area. After visiting this area, you couldn't help but feel positive about the future for the Apple II line.

Ms. Korman, a teacher from Los Angeles Unified School District, had an office set up where she demonstrated using her IIe and a program for automating IEP (Individualized Educational Plans). Anyone who has ever had contact with the paperwork involved in teaching special education is aware of the need for a computerized system. Each day, forms need to be completed, goals have to be charted, and detailed records must be kept on each child. People in attendance appreciated Apple inviting Ms. Korman. They had learned some valuable information to bring home to their staff.

The science environment was another popular station, and the two teachers Doug Kirkpatrick and Shelley Stocker were almost like disciples. They expressed how their teaching had improved since Apples were introduced into their classrooms. Experiments involving heat, temperature, sound, motion... all were conducted using Apples. Their students enjoyed learning to the extent that many times it was hard to get them to leave the classroom.

Pilot Programs

The early education area captured eveyone's attention. Young students (K-3) were demonstrating some of the things they were doing in the pilot program in North Carolina. Barbara Secrest (Apple), explained how Apple came out to this school district, and provided the teachers with everything from hardware, to training. They literally set up an early classroom education program.

Barbara explained that the program was designed to be matched, multiple learning — tactile, visual, and sound. The Echo board speech synthesizer was installed to use products such as Talking Text Writer. Muppet keys were used for the tactile learning environments. They also provided training for the teachers and a manual, published by Apple, included objectives, ways to integrate the computer as a tool, activities and management suggestions.

It has been a very successful project — obvious to all who watched while the young students worked at the keyboards — and, interesting to contemplate, this extensive involvement Apple has with education.

Regarding User Groups

Ellen Leanse, Apple's User Group Evangelist, should be commended for the job she is doing, opening communication channels between user groups. During AppleWorld, she was always there, introducing speakers, making suggestions, troubleshooting. She even emceed the "*Wheel of Fortune*" game conducted between user group members and Steve Wozniak at the user group dinner Monday night.

It was **Jean-Louis Gassée**, however, who best described the attitude Apple seems to be adopting toward user groups:

"It's good to be here and be a part of you, the people who made us. And it is very special to be, as you are, a member of a user group. (I am a member of 2.) When I was an Apple fan, even before I joined Apple, I was seeing a certain magazine from a certain computer society in the northwest which French customers bought avidly — Call -A.P.P.L.E. *So, I became a member of A.P.P.L.E. Co-op and I always treasured the relationship our company maintains. I live somewhere between Cupertino and reality. You represent the reality. I value the feedback, the encouragement, the criticism. If you bitch, you bitch — I want to hear about it... because indifference kills."*

An Apple II Update

(Apple II panel discussion.)

Neil Amsden:
Chief Evangelist Consumer Products
Harvey Lehtman:
Apple II System Software Manager
Rob Moore:
Hardware Engineering Manager IIGS
Mike Smith:
Apple II Product Management
Tom Virden: *Moderator*
Consumer Product Manager
Rich Williams:
Software Product Engineering

Tom: We know that there's an overwhelming amount of information here about Macintosh, and that's obviously to be expected because we introduced two new Macintoshes... and we had our glory, oh, about five months ago. Since then, alot of good things have happened in the Apple II world, and I want to make sure you realize that there's alot going on behind the scenes — we'll talk about that today.

As Jean Louis said, over the last five months we've totally changed and upgraded the entire Apple II line. We've brought out the IIGS and the neat thing about the GS — what I think most of the people that were on the team agree on is — is that we came out with more in that product than most of you, the dealers and the people we talked to, expected from us. It had all the features, plus it had the compatibility which made us feel good when it hit the marketplace. We quietly came out with the upgraded IIc, with the memory card, which was one of the number one things our customers asked us for. You know, "why can't we expand the memory in the IIc?" So, we came out with that at the same time. And, as I'm sure you know, about a month ago we came out with the new Apple IIe in a different color and a new professional keyboard. That just zipped right into our product line and is doing very well and is well received by the dealers and educators that are using it. It's also nice to see that in five months we've really changed the entire product lines to platinum. We thought about this about a year ago and said... "Can we do this?"... I mean all these color changes and what about peripherals that don't match and things. But, if you look at the product view, it's pretty amazing to see everything in platinum for the first time and I think they look very impressive. So, alot of things

have been going on, on both sides of our product lines.

We have with us today the people that really made those things happen behind the scenes. I'd like to introduce those people to you — we've got alot of talent from the management of the Apple II family and successors here today.

The first person I'd like to introduce is Rob Moore. Rob, and many of you may know him, is the hardware engineering manager for the Apple IIGS, and now the engineering manager for all the new features of the Apple II's. He resided over the six custom chips in the GS. Rob is here to answer some of the hardware questions you may have.

Harvey Lehtman is definitely a long time Apple Veteran. He's in charge of system software for the Apple II line. He's done the ROM on many products. On the IIGS they did an entirly new operating system, going from an eight to a sixteen bit operating system, as well as those incredible new tools to do the Apple user interface, and brought those all out in an incredibly short amount of time as Harvey will attest to.

Next to him, we have Neil Amsden who takes the word to the developers. He's our chief evangelist for our consumer and education, new third party products. Neil will be able to talk to us about what's going on within the development community. What they're doing with the tools, what products they're developing — to the extent that he can talk about them — and how they're receiving some of the features that we've put into both the current Apple II's the eight bit Apple II's and the sixteen bit Apple II's.

The final person we have is Mike Smith, who has brought out the two new IIe's and IIc's in the last three months. He's been very, very busy behind the scenes. He's the head now of Apple II management at Apple and he can talk about what we did in the IIe and IIc to upgrade them, and some of the features we're working on — those he is at liberty to discuss.

The final thing I want to talk about is that we really want to get some of your feedback — where we need to go next. Since you people implement behind the scenes on the Apple II's, and kind of stand for the words "*Apple II Forever*," we'd like alot of feedback from you as to where we should be going with those products in the next couple of years.

I'd like to start off with a couple of questions to get everyone up to speed here, about some of the things that are going on. The first question I'd like to ask is for Harvey to give us an update on where the Apple IIGS system software is right now, and some of the things they're working on. I'm sure you have a few questions on that.

Harvey: Ok... first I'd like to introduce Rich Williams. Rich actually does ROM's — I just sort of whip people into doing ROM's. Rich has been at Apple since... what? It's eight years now, and probably knows more about Apple II's than anyone else at Apple. So we're going to call on him for some questions too.

Where we're going right now on system software? We're obviously not finished with quite a few things. We're continuing to evolve the toolkits on the GS. In particular the sound toolkits are continuing to get layered. We're making available a note sequencer to developers with the system disk evolution. Developers have that right now. We're making available evolving versions of the ProDOS 16 operating system and we're going to the new version of the ProDOS 16 operating system — version 2. It will be made available to developers, although it probably won't be released on a system disk until early next year. The *Apple Programmers Workshop* is getting released in packages through *APDA*. I hope by now everyone here knows what APDA is. (I see one negative nod.) APDA is the Apple Programmers and Developer's Association which is run by the A.P.P.L.E. Co-op people up in Seattle. And it's a vehicle through which we're making available technical documentation for the Apple II and Macintosh lines, as well as third party documentation. We're also making available programming tools alot earlier than we can get it out through dealers. So, for instance alot of the earlier developers for the GS received waves of written documentation that changed very rapidly. Neil is responsible for sending alot of that out. I think by the time the last box went out it was, like 77 pounds of material.

Neil: "I just don't want to know how much we spent on shipping. That's all I'll say".

Harvey: In any event, even though we sent that out to alot of the developers, we couldn't send it out to everybody. We are going to make alot of that material available through the Addison-Wesley series of books. But the lead time for the production of those books is rather long. APDA is going to be able to make that information available... uh...

Neil: Now...

Harvey: Now!

Harvey to Neil: In fact, can you say a little bit about what's going on there?

Neil: Basically at this point... as far as I know, virtually all of the technical documentation suited for APW, for the GS in general — Firmware Reference manuals, Toolbox Reference and so forth — all that, is available through APDA right now, and they... (in frustration Neil hesitates, quickly exhales, and continues)

Those of you who have probably worked through APDA, in fact, probably all of you may have, at one point or another, encountered some, some... some problems there initially. Like any new organization, it was stretched pretty quickly. In fact the demand for APDA's services really exceeded what we expected initially. APDA now has something like 10,000 members which is tremendously positive, but has caused some problems.

They were not set up initially to deal with that size of demand for Apple IIGS specific information, as perhaps... uh... they should have been, but that has really been worked out. To the best of my knowledge they're now turning around orders in a 48 hour time period. So they really are responding quickly to orders for technical documentation. Everything is available through them, not just documentation, but also the APW assembler is available through APDA at this point. So they have a very complete set of documentation and tools available now.

Harvey: We're expecting the APW C through APDA very soon.

Neil: I don't have a definitive date on that one right now.

Tom: Neil, while we're on the subject of developers, can you give us a little bit of an update on what's going on in the evangelisim world — different products, the people you've been talking to?

Neil: Sure. I think in general, my reaction to where we are right now is, is... I'll be honest... it's a little mixed

because you build up such high expectations in the process. The quantity of development that is under way from the GS right now, is substantial. And I'm very, very, happy with it. In fact, I think you will be very happy with the results of that development in the course of the next three, six, nine, months.

The pace of the development has been a little bit slower than perhaps, I would have guessed, and would have hoped for six months ago. I think there's alot of factors involved in that, and, it encompasses everything from the state of our own development tools, as I'm sure Harvey will acknowledge, the state of the development environment, the C, the Pascal available from TML, the assembler... to higher expectations on the part of the developers as to what they felt they could accomplish. Ah... A little bit of everything. I think one of the biggest factors has been a realization that programming for this machine is programming for an entirely new machine. It's not, — most of the companies who initially set out to develop applications for the GS, looked at it... *oh, it's an Apple II*, — so we will devote resources to this machine. Nothing wrong with that, but they should learn to develop for the GS. But, the learning curve involved in that process is very similar to what developers underwent to begin developing for the Macintosh when it first came out. It is an entirely new type of programming experience and alot of the developers have found that to be somewhat more demanding than they had initially supposed.

We currently have... I think something in the order of fifteen applications shipping for the machine now. All told, there's about twenty five products. I'm including hardware products as well. There are many, many more applications and things under development. You begin to lose tab with it a bit once it becomes a public machine because you're not as directly involved with everyone who's doing development work.

Through the seeding process that we had in place while the machine was still under development, we placed machines with over a hundred and fifty companies. And part of the process was getting commitments from those companies to do development for us specifically for the Apple IIGS. I'm not going to sit here and promise that each and every one of those companies is indeed going to follow through on their promise — some will, some won't. But it gives you an idea of the scope of the development work that is underway. And, there are alot of companies, many of whom you have not even begun to hear from, and have not made any kind of public recognition of there plans at this point, developing applications for the GS.

So I think in general, the situation is very good and the feeling amongst the developers is very, very positive and there's several, — there's a couple of factors contributing to that.

The response that they are getting in terms of just demand, for GS development is very high, and obviously, that's going to drive their decisions to where they put there resources. Not only in terms of requests for GS versions of existing products, — but the earlier returns on some of the GS applications that are shipping, are very, very positive. And the companies who have begun shipping applications, indicate to me that they're very happy with the results, which of course pleases me, because that means that they'll turn right around and continue to develop more applications for the machine. So that kind of gives you a snapshot of where we are.

Tom: Before we go on, there's one thing I want to clarify. The question was asked of Jean-Louis in the last session about graphics on the screen. I don't think he understood the question. I think what he was talking about was our video chip that we've acknowledged. It's very obvious that there's a couple of cosmetic problems with it. To update you on that, we are shipping revised chips that work, — 100 percent, up to spec — and we will be announcing an upgrade program sometime in the near future to upgrade the people with the old video graphics chip. So, you'll be able to take care of it.

Harvey: I don't think it's any surprise to those of you who have a GS that there is a socket on that part.

Audience: While you're on the subject of chip problems... the write-protect ROM in the three and a half inch disk drives. Is there going to be any upgrade program for existing drives to get that chip changed or what?

Rob: I don't understand the question...

Audience: ...The write protect bug in the ROM...

Neil: Is anyone on this panel aware of a write-protect problem?

Rich: Yes, the problems with the early Liron drives — the early Unidisk 3.5's, some of the earlier ones, they did not give the write-protect status back properly. And what will happen is that if you attempt to save a file to a disk that has been write-protected, ProDOS first does a status call to the drive — says is the write protect tab on?

The drive say's no, no, no, it's not on. ProDOS say's ok... I'm going to start writing to the drive. Tries to write the first block of the file out there and then the disk goes, oh no! Now I'm write-protected! And ProDOS gets aptly mixed up from that and comes back with an error, and just, just isn't happy.

I know that the problem has been solved. And the corrected ROM's did go into manufacturing some time back. And, I do believe that all the drives we're shipping now do have the proper ROM's in them. What I don't know is what we're doing as far as having service done and upgrades on the drive. And you might do better talking to the dealer on that.

Audience: I have heard that the official position at Apple was that "we expect developers to work around that" — that problem. Is that not the official position?

Neil: To be honest with you... I don't think there is an official position Because this is the first time anyone in marketing has heard of this problem.

Rich: The official position certainly ought to be that if the operating system, which is ours, is not working right with the disk drive, which is ours... I'd say, the problem is ours.

Harvey: And I'll say that the official engineering position is that... we fixed it about a year ago. And indeed, I assume that it had gone into production. That was brought to my attention two months ago, but it should be fixed.

Tom: Let's start taking some questions about current Apple II's right now and in a few minutes I'll turn it over to getting some feedback from you. Let's take it from this gentleman right here...

Audience: What's going on with the tech notes? I know that there are several developers who have GS products planned, but they're waiting for the print docs.

Harvey: Ah... we have promised some of those developers to get a print manager to them for licensing in the very near future. I don't want to broadcast the date to all of you up here right now, but you will know as soon as it's released. It's actually very soon now.

It's a difficult issue. We wanted to be able to have the capabilities of the Macintosh print manager. We also had to provide color capabilities for the ImageWriter II. And we also had to provide a general enough version so people could write drivers for other printers as well. And that's where the complication came in. Ah... fourteen developers we believe, have a version in their hands.

We gave it to one developer to install and it will be available for licensing assuming that it's all working.

Rob: We've seen some demonstrations by two companies, finished products that they have... ah beta version, and they're pretty exciting.

Harvey: Right, but the the print manager is not firm enough that I felt we should licence it and it's going to be out within the next... (Neil Amsden begins speaking)...

Neil: Yes. That is our number one priority as far as what next to get to developers.

Harvey: And there is continuing evolution of the other tools as well. I want to point out to people that there was a new system disk that came out. We sent it out to manufacturing in December, and it is shipping with all the new machines. All dealers have been told that all of you who own GS's should be able to copy it to take advantage of the improved tools. And it's not the last one, it's an evolving machine.

Audience: Does it have a number?

Harvey: One point one. Repeating a question asked "Can it go to user groups?" Absolutely. It's copyable.

Neil: System disk... we want as high a distribution of it as possible.

Harvey: It can be used, but you can't sell it. In order to put it on your products, if you're a developer, you have to go through the licensing department, our licenses are very, very inexpensive, it's fifty dollars, enough to pay for the paper work. And, the important point about that, is, if there is the inevitable upgrade, and you've licensed it... you're notified and given the update.

Neil: Really, that's geared towards somebody who's developing software commercially.

Harvey: But, I have absolutely no problem, and I don't believe Apple has a problem with people reproducing the system disk for (their own) use.

Tom: Let's move on, we had some questions over here...

Audience: I'm running the Apple IIe, a Mac+, and an Apple IIc. Not yet a IIGS. I don't want to upgrade my IIe, I don't know what I'm going to do about it. What I'm looking for, is, the promised AppleTalk card which has been promised for two years for the IIe. When is that coming? Because I do alot... I have a lot of work I have to transfer through my modem back and forth between my Mac.

Tom: I think we'll address this question to, to Mike Smith.

Mike: I know, we discussed this earlier...

Audience: I just thought I'd try to catch somebody else and see if I can't get it.

Rich: Are you the manager of preannouncment? (everyone laughs)

Mike: No, no I think my answer has to be the same as it was earlier today, we're obviously looking at all sorts of different things, all sorts of new technologies and... ah... that may be one of them. (silence)

Tom:... Ok we had some questions... way in the back there.

Audience: Ever since AppleWorks 1.0 came out, it's had 12 print drivers installed. It's been a number of years, but we still have the same twelve printers installed. Now obviously you have custom printer, but it would really be nice if they would upgrade to some larger selling of printers that are out there... Some of the twelve that are listed on the AppleWorks, (menu). Especially having one that's for twenty four pin-driven printers with that new technology coming. It would be nice to address that, make AppleWorks a little more current than 2.0, which is a nice upgrade.

Tom: Ya, we're just about out of time. The next session here is Guy Kawasaki's software session. We're all actually with the personal sessions to talk about the hardware side. He'll have, I think some of his product managers there, and it's probably better for that group because none here are product managers for AppleWorks.

Harvey: However, they could take the hint too.

Tom: I'd definitely bring it up. Ok, a question back here.

Audience: All of the existing IIe software obviously addresses slot one as the printer port, and when you run that software on the GS and expect it to go out to the built-in port it doesn't work. Can't a driver be created that will automatically convert what would normally go out port one, out the GS serial?

Tom: This is a Harvey Lehtman question. (some kidding around and Rich Williams fields the question)

Rich: Actually it's surprising that you're having any problems with that. The way that the Apple IIGS works on its built-in ports is that it actually does have firmware, which you can enable via the control panel to be in slot one, which emulates the Super Serial card. So, most software that we have found, you know, that does expect to talk to slot one, works fine, and there isn't any problem like that. The exception to that, taking back my earlier statement of only a few seconds ago, is that some hardware, or... some software (rather) most notably, some software which emulates terminals, they have real problems. This is because they don't use the built-in firmware. What they do is talk straight to the hardware, which is a 6551 ASI in the case of the Super serial card — which they expect to be in slot two. And, in order to provide AppleTalk on the IIGS, we're using a different serial chip, which actually is mapped to different addresses and that's why these older terminal programs won't work. You might find that if you have another program, which is maybe a screen dump program or something, it might be running into a similar problem. But most of the packages, like AppleWorks and stuff, work fine. Using... (interrupted)

Audience: AppleWorks and Apple Access — but I thought the majority of the IIe software that was created before the IIGS, would work?

Rich: Well, most of it does. The main exception is terminal programs. If you're using something like... ah... like Access II or something like that, that will have problems. And the reason why they didn't use the firmware is, you know, because in order to provide a terminal that runs 9600 baud, and eighty columns on the screen, and try to do scrolling and what-not like that, they just couldn't afford the time to go to the firmware. But... most things do work fine.

Audience: You set the control panel for serial port or slot one.

Rich: Right. If you set the control panel for the printer port to be in slot one, then, that will mean that our built-in firmware is turned on.

If you set it for your card in slot one, that means our built-in firmware is turned off. And your software would not recognize the built-in printer port as being there.

Audience: (New) AppleWriter... AppleWriter, Apple's own... AppleWriter you gotta do a three byte patch in order to make it work, right? (long silence then Harvey speaks up)

Harvey: AppleWriter... AppleWriter is a four year old product.

Tom: Ok we're getting into some software questions again. Rob Moore hasn't been able to answer any hardware questions yet. I've never heard Rob so quiet this long before in my life, so somebody get some hardware questions here. We have a question back here...

Audience: I was using a program (on a IIc) put out by Schwab, and we did have a problem getting systems on that. The only fix was putting in a brand new, old style IIc motherboard. And, if you have people out there... I mean, I ran into this through a user group. This guy was ready to use his IIc for a football.

Neil: I don't know what the fix in Schwab's case will be, exactly. I can tell you that I know for a fact, they've been in touch with our tech support department. They're aware of the bug. We're aware that it causes problems with their program. So I mean, the communication's there, the resolution may not be.

Neil: (In reference to confusion about what program was having what trouble) There's a program from Carl Schwab, it's a communications program that demonstrates a problem with the new ROM's. It also has a problem with the debugging ROM's... Equalizer. (name of program)

Tom: We've got about ten minutes left here. One of the things that we'd like to do is get some feedback from you. I know none of you have any opinions about our products and we should be going. (laughter)

And so, you know, what do you want to see in the next Apple IIc... IIe... IIGS..

(overwhelming response) Laptop! (some laughter and then someone speaks out)...*Port-a-ble!*

Tom: You want a laptop AppleTalk. (laughter)... Can we define laptop a little more clearly? Just, you know, the specific features of that laptop.

Audience: I want nothing too complicated...

Tom: How many lines on the screen?

Audience: Eighty... No! twenty-four. That's adequate but not great.

Tom: You want a full page screen?

Harvey:... How much money? (laughter)

Audience: Two Thousand...

Tom: You're willing to pay a premium?

Audience: Yes! You get a good quality machine... Fifteen hundred.

What about a Personal Laserwriter?

Tom: Personal Laserwriter?

Audience: (in reference to personal laserwriter) Laptop! (laughter)

Tom: This can be defined as how many dollars? (laughter continues)

Audience: Ah...two thousand dollars.

Tom: Ok... PostScript?

Audience: Yes.

Tom: AppleTalk compatible?

Audience: Yes...

The discussion was running over and had to come to a close. Guy Kawasaki, and several software developers, took the stage. The discussion that followed involved mainly Macintosh products. For more information on the Macintosh aspects of AppleWorld, see April *Mac-A.P.P.L.E.*

In closing our coverage of AppleWorld '87, everything about it gave us confidence in the future... Confidence that user groups would be listened to, confidence that both lines of computers – the Apple II and the Macintosh — will progress into the future.

Please rate this article on the Reader Service Card by circling:

101 Excellent **102** Good

103 Fair **104** Poor

Blankenship BASIC

$39.95 For APPLE II+, IIe, and IIc

ProDOS and GS 3.3 versions

MONEY BACK GUARANTEE

either version **$25**

add $2 P&H ($5 outside U.S.)

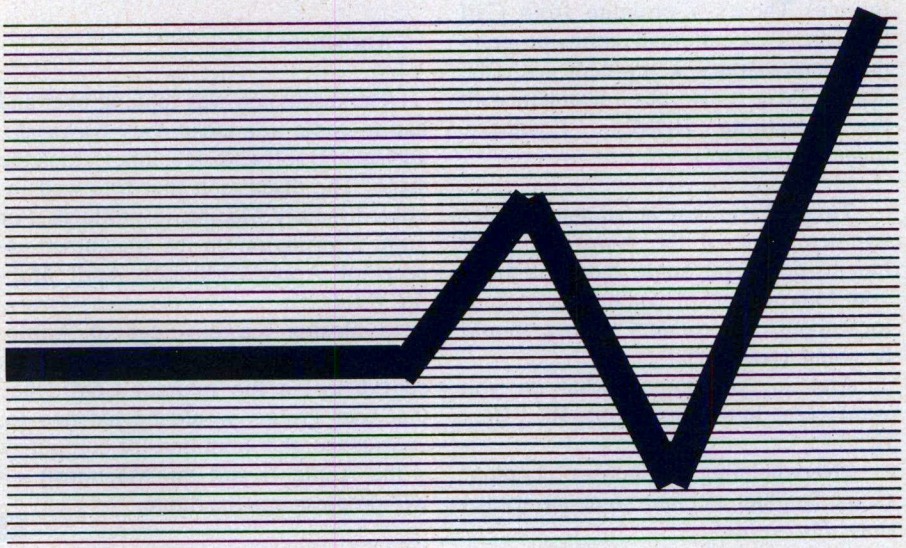
ONE COMPATIBLE PACKAGE

1. Real interpreter, not a pre-processor
2. WHILE-ENDWHILE and REPEAT-UNTIL Loops
3. True IF-THEN-ELSE-ENDIF (using WHEN)
4. PRINT.USING, FILE, MERGE, RANDOMIZE
5. PRINT and TAB commands work in HIRES
6. 80 columns supported on IIe and IIc
7. Full editor with AUTO-NUM and RENUM
8. Fast SORT, SEARCH and INSTR$ commands
9. BOX, BOXFILL, DRAW.USING and SOUND
10. Listings are indented automatically
11. DISK command replaces DOS'S CHR$(4)
12. DEFINE and PERFORM named procedures
13. 99% upward compatible with Applesoft
14. All commands entered normally, no &'s
15. 100's of satisfied users world-wide
16. FREE newsletter available to owners
17. Makes your Apple into a NEW machine!!

Mail CHECK to John Blankenship
P.O. BOX 47934 Atlanta GA 30362

CIRCLE **6** ON READER SERVICE CARD

A.P.P.L.E. Doctor



Bruce F. Field / Robert C. Pratt

Using Mac Drives

Gary Little had an article in the July 1986 *A+* explaining how to use the Macintosh 400K drive with the Apple IIe using Central Point's Universal Disk Controller. With the upgrade of the Mac to 800K drives, there should be plenty of reasonably priced 400K drives available. The only problem is that there is no eject button on the Mac drive (unless you're handy with a paper clip), but Tom Weishaar published a short routine in the June 1986 *Open-Apple* which ejects the disk.

Compatibility

Q. *I am considering purchasing an Apple* IIGS, *however, I don't relish having to give up some of my valuable software and hardware. I would like to know which — if any — of them might be compatible with the* IIGS. *Years ago, I purchased a Microsoft Z-80 card and have been writing and running programs in Microsoft Basic under CP/M. I would appreciate your comments about their possible compatibility with the* IIGS.

A. Some Z-80 cards designed for the Apple IIe (or II+) use non-Apple conventions on the I/O Select and Device Select lines in the expansion I/O slots. Among others, cards from Microsoft and Digital Research will not work on the IIGS.

Generally, interface cards that use interrupts will not work properly on the IIGS. Most standard printer interface cards or serial interface cards will probably work. But, the only safe way to be sure is to try it out.

ProDOS Shape Table

Q. *I am doing some engineering type programming on my Apple IIe with Applesoft screen BASIC. One difficult problem that I have run into is that I want to label items within my graphics. This may sound familiar. One way to do this is to use a shape table of the keyboard characters. Do you know where I can obtain a ProDOS shape table for the keyboard characters? I would like an additional explanation on how to use it. In addition, I was wondering if it would be possible to store a whole screen of graphics, on disk, to be recalled and displayed faster than having to regenerate it later?*

A. Because Shape tables of ASCII characters have been around for so long, and almost everybody has one, it may actually be somewhat difficult to find them in the stores. Beagle Bros' Apple Mechanic contains several ASCII shape tables and a shape table editor that you can use to create your own. Additional fonts are also available. Apple Mechanic programs are written for DOS 3.3, however, the shape tables are just binary files that are used by Applesoft not DOS, so a shape table on a DOS 3.3 disk can be directly transferred to a ProDOS disk using the ProDOS CONVERT program. You may need to load the table in a different memory location for ProDOS, because of the way ProDOS manages disk buffers. Generally for ProDOS you can load the shape table starting just above hi-res graphics page 2 at memory address 24576. Then, to protect it from Applesoft storing variables inside the shape table, set LOMEM to just past the end of the shape table.

It is easy to save and load graphics to disk. To save your completed graphics screen to disk, use BSAVE filename, A8192,L8192 for hi-res graphics page one (what you get if you use HGR in your Applesoft program), or use BSAVE filename, A16384, L8192 for page 2 (HGR2). To load it back directly to the screen, you will first want to turn on the graphics page with HGR or HGR2 so you can see it, and then load the binary file with BLOAD filename. No address (A) or length (L) specifiers are needed with the BLOAD statement if you want to load it back to the same screen it was saved from.

Detecting Printer Status

Q. *I would like to know if there is a PEEK that can detect whether or not a printer is active. My ImageWriter has a bad habit of 'going to sleep' after it has been on for more than half an hour. So my programs that involve printouts, just hang when they get to the printout routine. I have seen commercial programs (most likely machine language) which check and give a 'TURN PRINTER ON' or similar message, and I would like to do this from within Applesoft.*

A. There is no PEEK or simple test that can detect the status of your printer. The way you can tell what your printer is doing is to send it a character and see if it responds. If it is completely dead, turned off, not connected, or working properly, no indication will be sent back to the Apple. So, you can't tell for sure if it's on and working. The other possibility is that it is deselected (or acts like it is), and then when you send it a character, it sends a busy signal back to the Apple printer interface card. The software on the card continuously monitors the printer, waiting for it to become unbusy before continuing. This is what hangs up the Apple.

A possible solution is to bypass the software on the printer interface card and write a routine to send a character, and monitor the busy status of the printer. Exactly how this is done will depend on

your printer interface card. For the Apple Super Serial Card you should be able to POKE the character to address 49304 if your card is in slot 1, and monitor the status at address 49305. Unfortunately, that doesn't work because among other things there is a quirk in the 6502 that causes it to issue a false read which confuses the Super Serial Card. For this card, you would have to use machine language. An easier solution might be to get your printer fixed.

Passwords

Q. *How can I password-protect my AppleWorks files?*

A. The only fool-proof way of protecting your files is to encript them outside of AppleWorks and then decrypt them just before using the file under AppleWorks. This is more trouble than it is worth (unless you work for the CIA.) Super MacroWorks allows you to password-protect your AppleWorks Version 2.0 files. But, anyone using a different copy of AppleWorks can access the files from your data disk without specifying the password.

Try MacroWorks

Q. *How can I print a multiple column document in AppleWorks?*

A. You can use the spreadsheet under AppleWorks to type the columns and then move the document over to the word processor. As an alternative, MacroWorks has a column printing feature.

Memory Expansion Boards

Q. *I am an Apple II+ owner who is planning to upgrade to a IIGS. What memory expansion board can I buy now to work on both machines?*

A. Judging by their ads, either a RAM-Factor or Apple Slinky card should work on both the II+ and IIGS.

Color Monitor Problems

Q. *Have there been problems with fuses blowing on the Apple Color Monitor IIc?*

A. The original fuse used in the Color Monitor was too limiting. Apple is upgrading the fuses as the monitors are brought in for repair.

Using 88 Column Mode

Q. *How can I use the 80 column display mode from Applesoft BASIC?*

A. The IIe, IIc and IIGS have a built-in 80 column text display routine in their ROMs. To enable the 80 column firmware, insert the statement, PRINT CHR$(4);"PR#3". To disable the 80 column firmware, insert the statement, PRINT CHR$(21) or type <escape> <control>-Q from the keyboard. Do not use PR#0, because this will fail to reset certain flags. While the 80 column firmware is active, <escape>4 will shift to 40 column display and <escape>8 will return to 80 columns. On the original IIe (which, like mine, has not been enhanced), you cannot use column tabbing or HTAB while in 80 column mode. Instead of HTAB n, use POKE 1403,n-1 (where n is the column where the next item is to appear.) HTABing beyond column 40 causes non-screen memory to be overwritten.

If you want to protect your program from a user leaving the 80 column mode, use the GET command for input instead of INPUT. GET does not execute control character commands such as <escape><control>Q, while INPUT obeys these commands before passing a string back to your BASIC program.

Restoring Uppercase Characters

Q. *I received an Applesoft program from a friend who has a IIe, but I only have a II+ without an 80 column card. All of the PRINT statements display punctuation and garbage. When I list the program on my printer, the PRINT statements have lower case characters. How can I fix the program to run on the II?*

A. If only a few lines are involved, LIST each one, then use <esc>IIJ <space> to position the cursor at the start of the listed line. Move the cursor over the line using the right arrow and repeat keys to reenter the line. The II+ will automatically restore the line in all capitals.

Please rate this article on the Reader Service Card by circling:

111 Excellent **112** Good

113 Fair **114** Poor

Copies of articles from this publication are now available from the UMI Article Clearinghouse.

For more information about the Clearinghouse, please fill out and mail back the coupon below.

UMI Article Clearinghouse

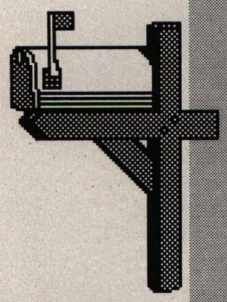
Yes! I would like to know more about UMI Article Clearinghouse. I am interested in electronic ordering through the following system(s):

☐ DIALOG/Dialorder ☐ ITT Dialcom
☐ OnTyme ☐ OCLC ILL Subsystem

☐ Other (please specify)_____
☐ I am interested in sending my order by mail.
☐ Please send me your current catalog and user instructions for the system(s) I checked above.

Name_____
Title_____
Institution/Company_____
_____
Department_____
Address_____
City_____ State_____ Zip_____
Phone (_____)_____

Mail to: University Microfilms International
300 North Zeeb Road, Box 91 Ann Arbor, MI 48106



Write-A.P.P.L.E.

Reader's Forum

Quit Call Comments

I was happy to see Rick Sutcliffe's "The Penultimate Quit Call" in the December 1986 issue of *Call -A.P.P.L.E.* The QUIT code is arguably the least attractive feature of ProDOS 8, and any attempt to gussy it up is most welcome. Unfortunately, I believe the article as printed contains one or two minor errors concerning ProDOS 1.2.

Through one of life's strange coincidences, I'd just disassembled the QUIT code in ProDOS 1.2 the day before purchasing the December issue. Thus, I was confused by Mr. Sutcliffe's contention that the ProDOS QUIT code turns the alternate character set on, when the code in front of me clearly turned the alternate character set off. To clear up this confusion, I disassembled the ProDOS 1.1.1 QUIT code and found that, indeed, the alternate character set was explicitly turned on. Hence, I can report that this "bug" has been eliminated — Apple Computer has apparently listened to its developers.

Something that may strike closer to home for users of the ProDOS 1.2 release dated 6-Sep-86, however, is that the first few lines of initialization in the QUIT code have changed. Thus, while the first twelve bytes of code in the ProDOS 1.1.1 QUIT code are:

```
AD 82 C0  LDA $C082
20 93 FE  JSR $FE93
20 89 FE  JSR $FE89
8D 0C D0  STA $C00C
```

as Mr. Sutcliffe reports, the corresponding twelve bytes in the ProDOS 1.2 QUIT code are:

```
AD 82 C0  LDA $C082
8D 0C C0  STA $C00C
8D 0E C0  STA $C00E
8D 00 C0  STA $C000
```

The result of this resequencing is that, unfortunately, Mr. Sutcliffe's routine to locate the QUIT code will fail with ProDOS 1.2. Replacing his search string with the twelve bytes above should help anyone who would like to install Mr. Sutcliffe's code in ProDOS 1.2. Since the program was published as a hex dump, I can't reliably report on just where the twelve replacement bytes should go. (I have sworn off entering programs which appear in strictly hex form — the checking and re-checking necessary to eliminate typos is simply too tedious.)

A number of other changes to the QUIT code have appeared with ProDOS 1.2. None of these changes affect the user directly, so I will leave their elucidation to another time. One particular change, however, is worthy of immediate comment. In ProDOS 1.1.1, the following code appears at the end of the main module: (Note: pseudo-ops presented are in ORCA/M form) (Listing 1).

All labels and variable names are my own. The intent of this code is clear — error conditions are checked after the system file requested by the user has been loaded and subsequently closed. No problem. Now look at the corresponding code in ProDOS 1.2 (Listing 2).

The 1.2 code appears to have some problems. As can be seen, the READ error status is never checked unless a CLOSE error has occurred. If a CLOSE error has ocurred without a READ error, then a random status register is pulled from the stack and brached on. While it would take a very peculiar set of circumstances for this bug to surface, one can imagine a particular sequence of bytes in the stack which could lock the system in a SKIP-to-BAD loop. More likely is the scenario in which the system file is successfully CLOSEd despite a fatal READ error. In this case, control will be passed to the fragmentary system program at $2000 — with predictably disastrous results.

I would appreciate hearing from others who have poked around inside of the ProDOS 1.2 QUIT code. Are there other versions of the code in which this bug doesn't appear?

Paul Harwood

Tidbits

In July, I purchased a 3.5" drive and upgraded my IIc motherboard. Three new

LISTING 1

```
        JSR     MLI
        DC      H'CA'           ;read the system file in
        DC      A'RD_FIL_DCB'   ;at $2000
        PHR                     ;and preserve error status
        JSR     MLI
        DC      H'CC'           ;close the system file
        DC      A'CLOSE_DCB'
        BCC     SKIP
        PLP                     ;adjust stack pointer
BAD     JMP     ERROR           ;and go to error handler]
SKIP    PLP                     ;no CLOSE error - any READ
        BCS     BAD             ;error?
        JMP     FILE_BUFFER     ;no errors
```

LISTING 2

```
        JSR     MLI
        DC      H'CA'           ;read file
        DC      A'RD_FIL_DCB'
        PHP                     ;preserve error condition
        JSR     MLI
        DC      H'CC'           ;close the file
        DC      A'CLOSE_DCB'
        BCC     SKIP
BAD     PLP                     ;get READ err. cond.
        BNE     ERROR           ;go to err handl. if READ err
        PLP                     ;else get rndm PSW from stack
SKIP    BCS     BAD             ;never taken if no CLOSE err
        JMP     FILE_BUFFER
```

motherboards later, I still have a computer that is not reliable. However, I've decided to live with it. When I read a letter in "A+" in January indicating I wasn't the only person with IIc problems, I decided to test the computer extensively. I spent 14 hours one Sunday doing that. For any disk the computer indicated had problems, I cataloged and verified it using Systems Utilities. I cataloged and verified over 50 (fifty) disks! Only 2 were bad!

Both ProDOS and DOS 3.3 disks behave erratically with this motherboard. Sometimes they'll work — sometimes they won't; sort of like telling your teenage son to take out the garbage. Maybe he will; maybe he won't.

Advice About Disk Problems:

1. Back up everything you can! (The motherboard also has outstanding disk-munching facilities.)
2. If your IIc is not behaving, turn it off and stand it in the corner. Come back an hour later. It just *might* have decided to cooperate with you. It has with me.

3.5" Disk Drive Problems

By all means, purchase the service warranty when you purchase a 3.5" drive. Mine has gone kaput 4 times in the 7 months I've had it.

If the 3.5 drive is booted, it will read from the built-in drive, but not from the external 5 1/2" drive that is daisy-chained to it.

If the built-in disk is booted, it will not read from the 3.5 drive. It will only read from the exterior 5 1/2" drive when the external drive is connected *directly* to the computer. In accepted Apple IIc style, daisy-chained to the 3.5 drive, it *cannot* be accessed.

The Menu

This Colorado firm has a marvelous new directory for Apple Programs (Cost-$10). They claim to stock 10,000 Apple titles. I tried one small order with them 2 1/2 weeks ago. Since it still hadn't arrived yesterday, I called them and asked where it was. (The order taker said it would be sent immediately, when I placed the order.)

I was informed that the normal delivery time is two to three weeks, and she implied it could "take longer if the program was not in stock." Sound familiar?

They say they charge the credit card when the order is shipped. I will check on that when my credit card statement arrives.

MacA.P.P.L.E.™

Information for your Macintosh needs

A.P.P.L.E. Co-op, who brings you *Call -A.P.P.L.E.* every month, also publishes *MacA.P.P.L.E.* for the knowledgeable Mac user. *MacA.P.P.L.E.* explores a wide variety of Macintosh environments through in-depth and candid coverage. Our user group roots allow us to provide a unique platform for issues and news important to Mac users.

Typical reviews and tutorials include:

• Business Productivity • Personal Finance • Desktop Publishing • Programming • Desktop Engineering • Graphics and Design • Artificial Intelligence

Featured analytical columns:

• **The Macintosh Revue** — a contrapuntal approach to the Macintosh marketplace. • **Mac Science** — how science is changing the Mac and how the Mac is changing science. • **So, What Else is New?** — trends and rumors. • **Silicon Intelligence** — parallels of human and artificial intelligence.

ORDERING *MacA.P.P.L.E.*

Co-op members: For the remainder of your membership period, you can receive *MacA.P.P.L.E.* as an additional publication ***for only $10!*** To take advantage of this offer, call our 24-hr. toll-free order line (see page 9) or use the order form on page 6. **Product Code: FMACAD.** *This offer is good through June, 1987.*

Non-members: A subscription *only* option is available for $21. Use the tear-out card on page 55.

The cataloge is well worth the $10. As for ordering software from them, I will think 4x before I call in the next order.

Cauzin

The Cauzin folks are excellent at giving help. My IIc motherboard problems led to great difficulties in using their stripper program.

Thought you'd like to know that the Navy is using Cauzin strips to send data on what supplies are shipped to a Navy ship. One of my ex-Navy sons says he wishes they'd had them when he was Active Duty — he was in Supply. Although he handled payroll, his buddies told him, at length, of the problem of entering supplies received into the ship's computer when the floppy sent with the supplies was crashed. In the Indian Ocean, it is a given that a needed floppy fresh from stateside won't work!

Now, Cauzin is busy at work writing a program that will allow "Dense" strips to be downloaded and printed on a Laser printer. A page of these dense strips will print in 20 seconds. This technology will be available to the Mac, and the IBM. At present, the Apple is not included: particularly because of problems like mine! We who have Apples *need* these dense strips! At least, for the GS! *Please tell Cauzin!! All of you!*

Broderbund

I love the Print Shop. However, when you are making a Sign, you can't line up the dollars and cents in the correct column. Try It! I called Broderbund, and they said I was correct — because of the "pixels" it can't be done!

Have you ever seen a sign where the prices didn't line up? I sure haven't! Here's the solution:

1. Design the sign without the prices. Before you print, write down which entry fields have entries, and which don't. Print the signs. (Be safe, and print 2 extras.)
2. Turn off the computer, and re-boot the Print Shop. Page to the written entry field. Using your list of fields, enter your prices, and justify them to the right side. Roll your paper back to the place the beginning must be to print the sign correctly. Print it — Eureka, it works!*When using this procedure, Don't justify the sign Top-To-Bottom.*

Another annoyance of the Print Shop is that it doesn't recognize a print buffer on a printer. I have a 128K buffer on mine. Because of this, my computer is totally tied up when I print out a lengthy banner. Grrrr...

PinPoint

Don't get me wrong — I love the Desk Accessories, and the Speller. However, why are they selling "Run-Run" for $49 when the "Toolkit" is $49 and contains "Run-Run"? I think if they sold "Run-Run" for $29 like UniDOS, they'd sell one to every Apple Enthusiast with a 3.5" or hard drive.

Infomerge — I'm bitterly disappointed with this program. It only works with one data base file at a time and not with a SpreadSheet file! ReportWorks has just had its price dropped to $49, and you can build a report using it that takes information from any number of data base or spreadsheet files. The limit is your Apple's memory! ReportWorks is difficult to use, but the folks at Megahaus are working on that problem.

Scotty Shipley

Repeating Characters

I have an Apple II that was converted to a plus. It has a hardware problem. Sometimes after a few minutes, other times after an hour or so, the computer acts as if I was re-entering the last key. It will re-enter at a rate that varies from every couple of minutes to several times a second.

I have loosened and re-inserted the cable that goes from the keyboard to the motherboard. Sometimes this helps, but eventually the problem returns.

Should I replace the cable? The keyboard? If so, what type of keyboard do I need? The old one is the pre-1980 mechanical type. Will the new ones work?

Larry Houston

The repeating characters problem you describe with your computer is most commonly attributable to intermittent connections somewhere between the keyboard and the motherboard. On a two-piece keyboard the cause is usually bad connections between the keyboard and the encoder subassembly which is suspended below the keyboard on twenty-five pins. On a single-piece keyboard it's harder to say. Try reseating the forty-pin encoder chip which resides on the keyboard circuit.

As for compatibility of II plus keyboards... absolutely. Some will not mount up to the underside of case without some difficulty. But there is one hundred percent electronic compatibility between Apple II and II plus keyboards. Please note that the IIe keyboards are entirely different. The encoder circuitry on a IIe is on the motherboard.

Swap Pins 2 and 3

The phone calls resulting from my "From IIe to and IIGS Beyond" article in the February 1987 issue have been quite interesting and varied. However, nobody has called me to chew me out about the mistake in the table describing how to wire up a cable to an external modem. Pin 3 on the modem should go to Pin 5 of the connector. Pin 2 on the Modem should go to Pin 3 of the connector.

For those readers wishing to hang me at sunrise, it has already been done locally. I spent the whole evening recently helping a local customer solve his problem. I have been associated with the telecommunications field for over 20 years and failed to apply my tried and true RS232C rule "If it don't work, swap Pins 2 and 3".

Working with this fellow, I also discovered some more confusion. The side connector in the manual represents the pin assignments for the female connector on the computer. My pin assignments are for the male connector. Looking at the male pins, the numbering is:

```
6   7   8
3   4   5
  1   2
```

Sorry for the inconvenience.

Cecil Fretwell

Analyza

I have been interested in reading your articles on "Silicon Intelligence" over the past few months. Something in the November article caught my eye. In college, I used the university's mainframe to play with a version of the "Eliza" program originally written by Dr. Joseph Weizenbaum. Since then I have been looking for a copy of the program which would run on my Apple II+.

Can anyone tell me where I can get a copy of the "Analyza" program mentioned in Silicon Intelligence, or another version of that program?

Clair F. James
c/o Call -A.P.P.L.E.
290 SW 43rd St.
Renton, WA 98055

THE MARKETPLACE

The following items are selected by our editorial staff from the large number of product announcements and news releases sent to us daily. They are presented solely to keep you informed about the current market for Apple related hardware and software. As we have not tested many of these products, appearance here does not necessarily represent endorsement or recommendation by **A.P.P.L.E. Co-op**. Further information about items appearing in **The Marketplace** can be obtained through the Reader Service Card. If you contact any manufacturer directly, please mention that you learned about them in *Call-A.P.P.L.E.*

THE NEW APPLEWORKS

Retaining the clarity and step-by-step instructional approach that made the first edition so popular, *Using AppleWorks,* 2nd Edition, explains the new developments in AppleWorks and in hardware. Two new chapters show how to move information among the AppleWorks programs and how to effectively use the Mail Merge program, which is new with Version 2.0. Using AppleWorks 2nd Edition features:

- Each chapter concludes with several pages of practical tips and specific applications.
- The Word Processor, Data Base, and Spreadsheet programs can be learned independently.
- Each example is extensively illustrated with pictures of the screen at each point.
- Many new screen displays have been added to the 290 displays already in the first edition.
- The Integration chapter explains important procedures not described in most other books or in the AppleWords documentation.
- "At a glance" appendixes include explanations of how to make AppleWorks work with a hard disk, how to Spell Check and other accessory programs, and even how to make a nonstandard printer work.

Price $17.95, *QUE Corporation, 7999 Knue Road, Suite 202, Indianapolis, IN, 46250, (800) 428-5331 or (317) 842-7162.*

CIRCLE **15** ON READER SERVICE CARD

A LIFEGUARD

Harbor Software, Inc., announces a new software package called LifeGuard for the Apple IIgs. LifeGuard is a program which protects computer users from accidental loss of valuable data by providing the ability to recover disk files that have been deleted under ProDOS 16 and ProDOS 8. The program has advanced features not found in any other program of its type. LifeGuard, using the GS mouse and a Macintosh like interface, can search an entire disk for missing files using full or partial names given by the user. Matching files are found anywhere no matter what directory they exist in. This ability alleviates the necessity for familiarity with directories and subdirectories. Furthermore, LifeGuard provides the capability of searching the contents of an entire disk to match on words or sentences. With this capability, the person who accidentally looses a file can find it based on its contents, and not be required to even know its name. LifeGuard is a major advance in disk utilities. Its powerful features and ease of use can prevent a major disaster.

LifeGuard is written entirely in 65816 assembler language for high speed performance and uses the Apple IIgs Toolbox. The programs strictly conforms to the Apple User Interface Guidelines for software design and used the Super Hi-Res graphics capability of the Apple IIgs. It is compatible with all ProDOS disk drives. LifeGuard will be available in May and will retail for $59.95.

Harbor Software, Inc. 403 Great Road #8, Acton , MA 01720 (617) 263-1870.

CIRCLE **16** ON READER SERVICE CARD

MOUSE CLEANER

Ergotron, Inc., announces Mouse Cleaner 360, a new computer mouse cleaning system designed to clean the mouse input device used on many personal computer systems and keep it operating at peak efficiency. Mouse Cleaner 360 safely and effectively cleans the tracking rollers inside the mouse to eliminate jerky movements and slow mouse response. Because of its unique, patented design, cleaning your mouse takes just a few seconds instead of minutes.

The Mouse Cleaner 360 Kit comes complete with a Velcro covered Scrubber Ball, Scrubber Board, lint free Chamois cloth, and bottle of Cleaning Solution. After applying a few drops of cleaning solution to the scrubber ball and inserting it into the mouse, the mouse is rotated in a circular motion over the scrubber board. This activates hundreds of tiny scrubbing bristles which whisk away gum, lint and other unwanted residue from the tracking rollers inside the mouse. Once the scrubber ball has done its work, the mouse cavity and positioning ball are cleaned with the chamois cloth and the unit is reassembled. In a matter of seconds, free cursor movement is restored. Mouse Cleaner 360 is compatible with all major personal computer mice which use a 1" diameter positioning ball including: the Apple II, IIe, IIc, IIgs. Price: $16.95.

Ergotron, Inc., 1621 East 79th Street, Suite C-133, Bloomington, MN 55420, (800) 328-9839.

CIRCLE **17** ON READER SERVICE CARD

MOCKINGBOARD MUSIC EDITOR

Electronics Realities, Inc., announces Mockingboard Music Editor. It is a music development utility which enables the addition of Mockingboard enhanced music to your programs with a minimal reduction in execution speed and useable memory. The Music Editor runs on any 64K Apple II+ or IIe; a Mockingboard A, C or compatible card is required for music output. Its main features are:

- Plays synthesized music on any Apple II+ or IIe with a Mockingboard A, C or compatible card.
- The software and manual costs only $5 dollars. Unless, of course, you obtain a copy from a user group or friend for free which we also encourage
- Specifically designed for use in commercial quality software applications.
- Up to 12,000 notes or 2,000 lines of musical code per score.
- Standard note durations are Triplets, Dotted, Double dotted, Whole, Half, Quarter, Eighth, Sixteenth and Thirty-second notes.
- Fine tuning feature allows customization of Tempo, Volume, voice characteristics, Tone values of notes, Duration of Octave and Accent effects.
- Range of 8 octaves – 3 below middle C and 4 above.
- 10 preset voices, 6 of which may be played simultaneously.
- The ability to create custom voices.
- Timbre effect.
- 6 sound channels enabling six notes to be played simultaneoulsy.

- 7 tempos ranging from very slow to very fast.
- 6 different volume settings from pianissimo to fortissimo.
- Tempos, volume and time signatures may be changed as often as needed within a score.
- Many musical symbols recognized, including sharps, flats, naturals, ties, octave markers and repeated passages with differing measures.
- Users may adjust most features, from the tone value for each note to the speed for each tempo.

The program and a 16-page manual may be obtained directly from us for a ($5.00) fee. The software without a manual may be obtained from many local Apple User's Groups.

We are also available to give technical support to any developer who may have difficulty working with the music routines, or to do some customizing of the program for particular projects.

Electronic Realities, Inc., 1115 Greendale road, Sheboygan, WI 53081.

CIRCLE **18** ON READER SERVICE CARD

MUSIC LIBRARY

Weekly Reader Software announces Music Library1 which provides 39 new pre-recorded songs on disk. There is a wide variety of music, everything from classical and folk to spirituals and childrens songs. You can play the music "as is" or modify and edit with the Stickybear Music program. A small sample of the songs you can play with this program includes an adaptation of Beethoven's Ninth symphony, The Battle Hymn of the Republic, On Top of Old Smokey, Silent Night, Camptown Races, Frere Jacques, and many, many more. This product works on the Apple II, II+, IIe and IIc with 48K and DOS 3.3. Sugessted Retail Price: $39.95.

Weekly Reader Family Software, 245 Long Hill Road, Middletown, CT 06457, (203) 638-2400.

CIRCLE **19** ON READER SERVICE CARD

DO-RE-ME

Multisoft announces DO-RE-ME, for the Apple IIe, and IIc users. DO-RE-ME consists of three full-function Subsystems: Electronic Spreadsheet (255 x 255), Word Processor, and a Data Base Management Subsystem (64 fields, max. record length 2048). Also integrated with DO-RE-ME are:

- Report writer.
- Mail Merge program.
- Desktop Accessories.
- Context-sensitive On-Line Help facility.
- Pop-up menu-driven user interface (similar to Lotus 1-2-3). makes this program easy to use.

The DIF/ACSII file import/export facility is also provided, should the user need to move data back and forth between external programs and DO-RE-ME. Files created by VisiCalc, The Spreadsheet, MagicCalc, and FlushCalc can be read directly, without any loss of formulas. Data can be integrated with text by passing a file from one Subsystem to another, or by using the DO-RE-ME Clipboard, without any data conversion and disk activity.

DO-RE-ME provides 35 text-formatting embedded commands, and 38 built-in math, business, date, and logical functions. These functions support date arithmetic, and accept character strings as input, and return strings as output. Most importantly, they can be specified not only in the Spreadsheet environment, but also in computed fields, in DBMS. DO-RE-ME supports up to 1 meg of auxiliary memory, hard disk, multi-volume files, and a set of DOS commands that can be executed without ever leaving the DO-RE-ME environment. Price: $49.00, plus $5.00 shipping.

Multisoft, 120 East 90th Street, Box 5J, New York, NY 10128. (212) 534-4047.

CIRCLE **20** ON READER SERVICE CARD

BASIC ALGORITHMS

Lambsoft Development announces Practical BASIC Algorithms, a tutorial/utility package designed for intermediate level BASIC programmers. It may also serve as a subject matter preview for people planning to take a computer programming course. Among the topics covered are stacks, queues, Shell sorting, binary search, binary trees, structured programming, a database manager and much more. The disk provides on-screen examples of how data is manipulated by the various algorithms as well as source code that may be incorporated into the user's programs without licensing fees. The documentation details how the algorithms work, the programming situations in which they are used and likely bugs. Extensive information on Apple II hi-res graphics, graphics demos, and shape and character font makers are also included as are seven complet hi-res character fonts, three of which show proportional spacing.

For Apple II series, 48K and one disk drive required. Price: $19.95 (+ $3.00 shipping and handling). Users who obtain the disk as shareware may obtain the documentaion and register for update information by sending $15.00 .

Lambsoft Development, PO Box 42801, Philadelphia, PA 19101 (215) 438-3695

CIRCLE **21** ON READER SERVICE CARD

SPELL CHECKER AND PICTURE MANAGER

StyleWare, Inc., announces the release of MultiScribe Spell Checker, a spell checking program designed for on-line use with StyleWare's MultiScribe 2.0 word processing program which allows the user full editing and spelling correction capabilities.

The MultiScribe Spell Checker comes equipped with a 50,000 word dictionary that allows a user to search, highlight, and correct one or more occurrences of misspellings in MultiScribe word processing documents. The dictionary can be easily expanded by adding new words, special spellings, and abbreviations to the user's personal dictionary. MultiScribe Spell Checker will even scan word lists for alternate spellings. Price: $39.95.

StyleWare, Inc., 5250 Gulfton, Suite 2E, Houston, TX 77081, (713) 668-1360.

CIRCLE **22** ON READER SERVICE CARD

ELECTRONIC HANDBOOK TOOL

ABRACADATA announces Electronic Handbook Tool for everyone who has any relationship to any Organization and has an Apple II series computer with 64K or more.

'Welcome!' is a new tool for any organization, whether it's a business, club, retail establishment, or school. It is a highly structured data base with both text and graphics. It enables the key person(s) in an organization to provide and update information for both new and current members to use and refer to. It includes these formatted file types:

Directories, Phone lists, Mailing lists, Job titles and Functions, Interrelationships, Organizational History, Goals, Emergency Procedures, Business Plan, Phone/Addresses, Orientation, Safety/Benefits/Insurance, and a user defined category.

The graphics file types include Building, Lot, Parking lot, Conference, Location Map, Plan, Org.Structure, Components, Process, Flowchart, Graph/Chart, and user defined.

The person in the organization who is the "Welcome! manager" uses the Manager disk to enter appropriate information about the organization. The manager then provides each user with a 'Welcome!' User and data disk to access the information. Additions and changes to the information are done only with the Manager disk and distributed to each user (or user station) as data disk updates. Each user can read, sort, search, and print all information on the data disk, including formatted reports and mailing labels. However, only the "Welcome! manager" can make changes or additions to the data.

The program has what we call an *enhanced* mouse interface with icons and pulldown menus. Since users have found that keyboard commands are often more convenient than mouse movements, we have enhanced mouse use with keyboard choices so that you can mouse select icons but you needn't move all the way across the screen to do so (unless you want to).

You move to any spot that is horizontally opposite the icon you wish to choose, and press <RETURN>. No buttons, no long

mouse journeys. After the section, the cursor remains where it was when you pressed <RETURN>. The program also works with the keyboard alone, or with paddle/joystick/graphics pad peripherals. Added features: 'Welcome!' can use project data from our Everybody's Planner project management package. Price:$79.95.

ABRACADATA, PO Box 2352, Eugene, Oregon, 97402, (503) 342-3030.

CIRCLE **23** ON READER SERVICE CARD

LOGO DATA TOOLKIT

Terrapin, Inc., announces the release of the Logo Data Toolkit. The Logo Data Toolkit, developed in conjunction with educator and Logo enthusiast Dorothy Fitch, contains several comprehensive programs for gathering, handling and manipulation data. The programs can be used *as is* or can be modified in the Logo environment to suit individual, classroom and curriculum needs. The programs include: a survey Tool for creating, modifying and reporting results of surveys, a Database Tool for organizing, editing and sorting data, an Associations Tool for setting up quizzes or playing matching games on related data, and Charts Tool for displaying data in pie, line and bar charts. Printing routines for each tool are provided.

The software is accompained by a manual containing descriptions of each tool program as well as helpful classroom activity ideas. An Integrated Projects section incorporates each of the tools in exciting projects including a look at advertising, studying weather patterns, exploring another country, and more. An Appendix contains information on Logo data structures, frequently used tool procedures and Data Conversions programs. Reproducible worksheet planning pages for each data tool are also provided. Terrapin offers versions of the Logo Data Toolkit for use with Terrapin Logo for the Apple (version 2.0 or later). Price: $49.95.

Terrapin, Inc., 222 Third Street, Cambridge, MA 02142, (617) 492-8816.

CIRCLE **24** ON READER SERVICE CARD

MICROBUFFER/MINI SERIES

Practical Peripherals introduces three new models of its Microbuffer/Mini series of stand alone compact printer buffers that feature both expanded memory and additional computer interfaces. The new models, designated the Microbuffer/Mini 512, come with a choice of either parallel to parallel, serial to parallel or parallel to serial interfaces, and are available with up to 512K of buffer memory, approximately four times the capacity of the previous Microbuffer/Mini 128. The Microbuffer/Mini family of pocket-sized, stand alone intelligent printer buffers eliminates waiting time while documents are printed. The Microbuffer/Mini 512 comes with 32K of memory, expandable to 512K of memory. The new models utilize either special 256K DRAM or standard 1 Mbit DRAM chips for easy user expandability.

The Microbuffer/Mini 512 models are part of Practical Peripherals' popular Microbuffer family of internal and external printer buffers. The Microbuffer In-Line, first introduced in 1981, is a stand alone unit available in either serial or parallel models for use with any computer/printer combination, and is one of the most popular printer buffer products on the market, with more than 500,000 units sold. The Microbuffer II+ family is a series of internal buffers for the Apple II, II+, and IIe personal computers. The Microbuffer/Mini 512 has a suggested list prive of $99.00 for 32K of buffering memory.

Practical Peripherals, 31245 La Baya Drive, Westlake Village, CA 91362, (818) 991-8200.

CIRCLE **25** ON READER SERVICE CARD

WHAT'S NEW IN GAMES?

War Games

Accolade Inc., announces "Power", a galactic war game "Power" is a 22nd century war game. The player is pitted against hostile Demoid Empire and his goal is to save Earth from terrorist attack. The demoids occupy the Earth's most vital Mitonium conversion plant, Earths source of power in the 22nd century, and demand an unconditional surrender. The player must infilitrate the occupied asteroid and disable the converter before it can be destroyed by the Demoids.

The player's mission is hampered by various Demoid battle ships. Using joystick controlled commands, the player has missles and light phasors on hand to attack or defend himself in galactic battle. The arcade style design of "Power" allows fast action and adventure. The Player's space vehicle starts out with a limited amount of energy which diminishes when shot by Demoid space vehicles and increases when traveling in the direction of the Mitonium plant.

When the space vehicle's energy supply is exhausted, the player must use his jetpack to go to a nearby pod where a reserve space vehicle is parked. There are eight of these pods throughout the game.

The player has to successfully complete seven levels of play before arriving at the converter. "Power" ends when the player completes his mission and disables the Mitonium power plant.

Featuring 5 skill levels. Other games available are Deceptor and Desert Fox Price: $14.95.

ACCOLADE INC., 20833 Stevens Creek Boulevard, Cupertino, CA 95014 (408) 446-5757

CIRCLE **26** ON READER SERVICE CARD

THE CHESSMASTER 2000

The Software Toolworks announces The Chessmaster 2000. Not only does Chessmaster 2000 pack in practically every feature, we've run across in other chess simulators, it also incoporates a new chess algorithm that makes it a top level player. Boasting the world's largest library of opening moves (over 71,000), this program works equally well as a tutor for beginners, an opponent for novice to expert players, and a crack analyst of your own game problem.

Beginners are offered every possible aid to learn chess, including a special mode that shows where moves can be made, various play hints, and a feature that lets you replay an entire game with a move-by-move analysis. As players improve, or for experienced hands, Chessmaster shows greater and greater expertise.

Great matches from history are on disk to replay, and there is a "watch the Chessmaster 2000 think" mode in which you can see the analytical functions of the program at work. The graphics are superb, with revolving 3-D and 2-D boards giving you views from the side and overhead.

Above all, there is the game itself, sophisticated yet easy to play with. Chessmaster 2000 is one of the year's best. Critics suggested age: 10+. Price: $39.95

The Software Toolworks, One Toolworks Plaza, 13557 Ventura Boulevard, Sherman Oaks, CA 91423, (818) 907-6789 or (213) 858-6789.

CIRCLE **27** ON READER SERVICE CARD

CHAMPIONSHIP BASEBALL

Activision announces Championship Baseball for Apple II series computers. Championship Baseball is an exciting baseball simulation which makes you team manager and player for entire season of big league action. You select each member of your team and put them in competition against the computer or a friend.

You can have your team play exhibition games to give them as much practice as you want. When your team is ready for some serious competition, Championship Baseball invites you to enter the 24-team, four division race for the pennant.

Championship Baseball delivers the best combination of graphics, realistic sound effects and gameplay on the market. The entire game of baseball is captured in this one simulation, from simple batting practice to a unique 3-dimensional splitscreen perspective that makes the game as realistic as possible.

Championship Baseball gives you total control. You begin by drafting your team of pitchers, relievers, infielders and outfielders from a pool of talent. Pitchers are rated on three separate skill levels (speed, control, and stamina); all other players are rated on four levels (batting, catching, running, and throwing). Once your team is selected, you set a batting order and play ball. Suggested retail Price: $39.95.

Activision, Inc. 350 Bayshore Parkway, Mountain View, CA

94043.(415) 960-0410.

CIRCLE **28** ON READER SERVICE CARD

MARBLE MADNESS AND APPLE

Electronic Arts announces the Apple II Series release of its award winning Marble Madness. Marble Madness is designed to appeal to a broad spectrum of gameplayers from the novice to the most experienced. In the one-player mode, using a joystick, you race your marble down treacherous raceways in an attempt to beat the clock to the goal line. In a two-player game, two racers compete head to head trying to beat their opponent to the goal line.

Six unique playfield raceways provide players with an ever-increasing challenge. Each level has its own "personality" and contains various obstacles and creatures to contend with. Players attempt to race through levels such as the Silly Level, where due to reversed physics, all previous rules are invalid, or the Ultimate Race, the game's grand finale. Other levels include the Aerial Race, where thrilling high speed jumps are the order, and a level which requires players to contend with a huge mechanical wave. Price: $34.95.

Electronic Arts, 1820 Gateway Drive, San Mateo, CA 94404, (415) 571-7171.

CIRCLLE **29** ON READER SERVICE CARD

II DESKTOP PUBLISHING

SpringBoard Software, Inc., announces Springboard Publisher program. Springboard Publisher offers users an unrivaled combination of global page control, a comprehensive set of text entry and editing tools, and extensive graphics library and precision control of graphics, even to the pixel level. For the first time, *complete* desktop publishing capabilities are available for the Apple II family of computers in one software product. The sophisticated desktop publishing program is designed to meet the needs of people who create and publish pages in many environments, including home, school, and business. Springboard Publisher lets anyone control the style and design of pages, no matter what their age or creative ability. Pages can be pre-formatted or the user can work free-form, letting ideas flow, trying different things and making changes. Graphics and text can be placed anywhere on the page and rearranged at any time, easily and efficiently. All of the tools are available on screen, accessible from easy-to-use pull-down menus.

The user never has to leave the page to set document parameters, enter text, add graphics or make changes. Both text and graphics can be entered on screen or imported from outside sources. Springboard Publisher is scheduled to be released during the 2nd quarter in 1987, at a suggested retail price of $139.95. It will be compatible with the Apple IIe, Apple IIc, and Apple IIgs personal computers. Requirements: The Apple version requires 128K of memory, a mouse, while optional, is strongly recommended. The program will support dot matrix and 24-pin printers. In addition, Springboard is offering a laser driver to support laser printers when used with the new program. The laser driver will be released at approximately the same time as Springboard Publisher at a suggested retail price of $39.95.

Springboard Software, Inc., 7808 Creekridge Circle, Minneapolis, MN 55436, (612) 944-3915

CIRCLE **30** ON READER SERVICE CARD

STANDBY POWER

Electronics, Inc., has used the latest in solid state circuitry to develop the new Scooter Model SPS45OVR Standby Power Supply/Voltage Regulator, rated at 450 watts continuous duty. Yet, it's housed in a unit one-half the size of a standard 450-watt power supply and offered at a very attractive retail price. Many experts feel that voltage regulation is just as important as backup power for protecting personal computers. Scooter answered this need by uniquely combining the two products into one compact electronic unit measuring 5 1/2" x 63/4" x 91/2: and weighing only 18 pounds. Also, the need for standby power supply protection with faster transfer times has increased with the growing demand for hard disk drives in personal computers. Model SPS450VR has an extremely fast transfer

JOIN

A.P.P.L.E. Co-op

World's Largest Computer Co-op

Get more from your personal computer investment!

Benefits of Co-op membership:

- **Become an Owner** - Co-ops are owned by their members, who receive annual dividends that correspond to their annual purchases times the percent of the Co-op's annual profit.
- 12 issues of :
  Call-A.P.P.L.E. magazine
  (Apple II/GS)
  and/or
  Mac A.P.P.L.E. magazine
  (Macintosh)
- **Hotline / Technical Support** : You can get information and help with almost any Apple-related problem from our staff experts. Our staff is backed up by a group of consultants which are specialists in a variety of areas.
- **Telecommunications** - The wave of the future that you can be a part of. With your Apple and a modem you can send and receive information, messages and programs from the A.P.P.L.E. Crate Bulletin Board Service.
- **Co-op Buying** - You will receive discounts (from 20 to 80 percent) on software, hardware, accessories and publications. Many items that members have found to be invaluable, are sold exclusively through the Co-op.
- **Public Domain Software** - Increase your software library for just pennies a program. Thousands of programs available for Apple II and Macintosh computers.
- **And much more.....**

Use one of the tear-out applications on the next page.

JOIN THE CO-OP TODAY !

New Article Rating System

Accompanying each article in this issue are four Reader Service numbers which you can use to rate your enjoyment of that article. Circle the appropriate numbers on the Reader Service Card at right. If you would like more information on products advertised, reviewed, or otherwise mentioned in this issue, please circle their corresponding numbers as well. Drop this card in the mailbox and you will receive the information you requested. Those members who rated the articles as well will be eligible for our monthly drawing at our Co-op meeting. **The winner will receive $100 credit toward any Co-op purchase. The top-scoring author will also receive $100 cash bonus.**

Advertiser Index

Reader Service No.		Page No.
1,2,3,	Applied Engineering	2,3,65
4	Blankenship Associates	53
5	Curtis Manufacturing Company	68
6	Diversified Software.	25
7	Macro Disk	36
8	Macro Solutions	21
9	MGR Software	41
10	Nite Owl Productions	36
11	Pergamon Software	13
12	Precision Data Products	14
13	Preferred Computing	67
14,15	Protecto /Computer Direct	34,35
16	Temporal Acuity Products	7

For additional products with Reader Service numbers, see The Marketplace and reviews.

READER SERVICE CARD

1	2	3	4	5	6	7	8	9	10
11	12	13	14	15	16	17	18	19	20
21	22	23	24	25	26	27	28	29	30
31	32	33	34	35	36	37	38	39	40
41	42	43	44	45	46	47	48	49	50
51	52	53	54	55	56	57	58	59	60
61	62	63	64	65	66	67	68	69	70
71	72	73	74	75	76	77	78	79	80
81	82	83	84	85	86	87	88	89	90
91	92	93	94	95	96	97	98	99	100
101	102	103	104	105	106	107	108	109	110
111	112	113	114	115	116	117	118	119	120
121	122	123	124	125	126	127	128	129	130
131	132	133	134	135	136	137	138	139	140
141	142	143	144	145	146	147	148	149	150

Please circle one answer for each item below

1. Are you an A.P.P.L.E. Co-op member?
   Yes No
2. What one computer do you currently use most?
   A. Apple II or II+ D. Apple IIGS
   B.Apple IIe E. Macintosh
   C.Apple IIc F. Other _____

Please indicate with a check mark which of the following is your primary source for hardware and software purchases.

	hardware	software
3. retail outlets	❑	❑
4. mail order services	❑	❑
5. on-line services	❑	❑

CA 5/87 • CARD EXPIRES 6/30/87

Name _____

Address _____

City _____ St _____ Zip _____

❑ *YES, sign me up* as a member of A.P.P.L.E. Co-op

$49 payment enclosed (includes $23 application fee and $26 annual dues)
Indicate which magazine(s) you would like to receive with your membership.
❑ Call-A.P.P.L.E. ❑ Mac A.P.P.L.E. ❑ Both Magazines (I'm adding $15)

❑ **Subscription only, please.** $21 for 12 issues - No Co-op membership
❑ Call-A.P.P.L.E. ❑ Mac A.P.P.L.E. ❑ Both Magazines (I'm adding $15)

❑ Please send me more information about A.P.P.L.E. Co-op.

❑ Please send me information about Institutional Co-op memberships.

❑ Check ❑ Visa ❑ MasterCard

_____ Exp. _____

Name _____

Address _____

City _____ St _____ Zip _____

Foreign postage rates available upon request

CA 5/87

❑ *YES, sign me up* as a member of A.P.P.L.E. Co-op

$49 payment enclosed (includes $23 application fee and $26 annual dues)
Indicate which magazine(s) you would like to receive with your membership.
❑ Call-A.P.P.L.E. ❑ Mac A.P.P.L.E. ❑ Both Magazines (I'm adding $15)

❑ **Subscription only, please.** $21 for 12 issues - No Co-op membership
❑ Call-A.P.P.L.E. ❑ Mac A.P.P.L.E. ❑ Both Magazines (I'm adding $15)

❑ Please send me more information about A.P.P.L.E. Co-op.

❑ Please send me information about Institutional Co-op memberships.

❑ Check ❑ Visa ❑ MasterCard

_____ Exp. _____

Name _____

Address _____

City _____ St _____ Zip _____

Foreign postage rates available upon request

CA 5/87

NO POSTAGE
NECESSARY
IF MAILED
IN THE
UNITED STATES

BUSINESS REPLY CARD
FIRST CLASS PERMIT NO. 217 CLINTON, IA

POSTAGE WILL BE PAID BY ADDRESSEE

Call -A.P.P.L.E.
P.O. Box 1993
Clinton, IA 52735-1993

NO POSTAGE
NECESSARY
IF MAILED
IN THE
UNITED STATES

BUSINESS REPLY CARD
FIRST CLASS PERMIT NO. 450 RENTON, WA

POSTAGE WILL BE PAID BY ADDRESSEE

A.P.P.L.E. Co-op
290 S.W. 43rd St.
Renton, WA 98055

NO POSTAGE
NECESSARY
IF MAILED
IN THE
UNITED STATES

BUSINESS REPLY CARD
FIRST CLASS PERMIT NO. 450 RENTON, WA

POSTAGE WILL BE PAID BY ADDRESSEE

A.P.P.L.E. Co-op
290 S.W. 43rd St.
Renton, WA 98055

time of only two milliseconds to meet these most demanding needs.

When primary power fails or drops below 15% of nominal, inverter power is automatically provided at a regulated 120v ± 5%. Your PC will continue running with a full load long enough for you to enable a normal shutdown. When primary power returns to within 7% of nominal, the system automatically switches back to normal line operation, and automatically begins to return the battery to full charge.

Mated to sophisticated electronic circuitry, a built-in maintenace-free, rechargeable battery delivers rated voltage to each of the four protected computer accessory sockets at the rear of the unit. The circuitry also monitors the line voltage for electrical spikes and surges, providing complete protection. Other features include an internal audible alarm reporting system ("primary power loss" and "low battery"), regulated automatic recharging, automatic shutdown to protect battery after discharge, alarm silencing button, status indicator LEDs, and master on/off switch. Price: $895.

Ohm/ Electronics, Inc., 746 Vermont Street, Palatine, IL 60067, (800) 323-2727.

CIRCLE **31** ON READER SERVICE CARD

DISK FILE

Curtis Manufacturing Company, Inc., has introduced its newest computer accessory product, the Curtis 5 1/4" File. Designed to hold 50 5 1/4" floppy disks, the Curtis 5 1/4" File features adjustable dividers with color-coded labels, a carrying handle, and anti-skid feet to anchor it securely on any desktop surface.

The Curtis 5 1/4" File is cast in beige plastic with a hinged, smoked bronze cover that includes a lip at the front to hold three disks in use during a work session. The built-in handle incorporates a push-button, nonmagnetic lock. Curtis manufactures and supplies computer accessories including printer stands, tilt-and-swival monitor bases, vertical CPU stands, cables, surge protection products and disk files. Price: $9.95

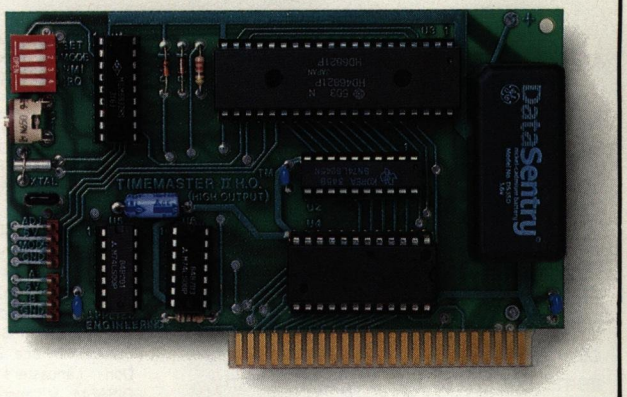
Curtis Manufacturing Company, Inc., 305 Union Street, Peterborough, NH 03458, (603) 924-3821

CIRCLE **32** ON READER SERVICE CARD

CONSULTANTS *Continued from page 9*

Logo (S. Australia)	Peter Carter	(08) 434298	
Macintosh (Midwest)	Tim Smith	(309) 734-7451	(C)
Macintosh (West)	Don Stinson	(206) 242-2590	(P)
Macintosh (East)	Dennis Baumgarte	(914) 856-2721	(E)
Macintosh (East)	Robert Forras	(516) 328-3433	(E)
Macintosh (East)	William P. Steinberg	(212) 921-0663	(E)
Macintosh (South)	Floyd Rathbun	(615) 865-3011	(C)
Mathematics	Stowe Keller	(804) 971-6876	(E)
Multiplan	Richard Schubert	(716) 352-0837	(E)
Multiplan (India)	Pravin Bhambri	660060 ext. 607	
Music	Marc Apfelstadt	(919) 282-1494	
Music	Stephen F. Lee	(704) 876-3444	(E)
Pascal (East)	Tom Woteki	(202) 547-0984	(E)
Pascal (Midwest)	Tim Smith	(309) 734-7451	(C)
Pascal (South)	Garry Flemings	(205) 279-7474	(C)
Pascal (West)	Paul E. Sleeper	(415) 494-1915	(P)
Pilot	Earl Keyser	(612) 429-5570	(C)
Pilot	Randy Warrens	(608) 837-9566	(C)
ProDOS/Apple IIe	Cecil Fretwell	(319) 236-0961	(C)
ProDOS (West)	William Reynolds	(503) 393-6758	(P)
Small Business/Cash Register comms	Michael Lipay	(412) 795-7673	(E)
Spreadsheet	Robert Greco	(609) 884-3047	(E)
Spreadsheets	Jeffrey Jacques	(904) 385-7947	(E)
Spreadsheets	Leon Raesley	days(301) 439-1799	(E)
Statistics/Educ	Rich Hofman	(513) 529-4420	(E)
Stock Market	Robert Wood	(703) 893-9591	(E)
Visicalc (East)	George L. Hefter	(617) 256-3082	(E)
Visicalc (North)	Vince Aderante	(312) 289-9685	(C)
Visicalc (South Africa)	Brian Abrahams	(Durban)315442	
Visicalc (South)	Dr. Bob Richard	(504) 466-0894	(C)
Word Processing	Leon Raesly	days(301)439-1799	(E)

The first clock card to add time and date to AppleWorks'™ screens and files.



Timemaster H.O. So many features it's become the world's best selling clock card.

It's the smart way to put the time and date on your Apple IIe® or II+®. Because only the Timemaster H.O. packs *ALL* the important features of all the competition *combined,* including leap year, year (not just in PRO-DOS), month, date, day of week, hours, minutes, seconds and milliseconds. It's totally PRO-DOS, DOS 3.3, PASCAL and CP/M compatible. And, of course, it works better than any other clock with AppleWorks.

With the Timemaster H.O. you're assured of reliable, trouble-free operation along with simple, built-in, one-command time setting, accuracy through leap years, and a firmware maintained interrupt setting.

If you're using or writing software for other clock cards, you're still covered. Because the H.O. will *automatically* emulate them. And only the Timemaster H.O. adds 15 new commands to BASIC. The H.O. even comes complete with two disks full of sample programs, including a computerized appointment book, a DOS dating program, interrupt programs, and over 30 programs that others charge you extra for — or don't even offer. As a low-cost option, you can add true BSR remote control to the H.O., giving you remote control of up to 16 lights and appliances in your home or office.

• Time in hours, minutes, seconds and milliseconds; date with year, month, day of week and leap year • 24-hour military format or 12-hour AM/PM format • Eight software controlled interrupts so you can run two programs at the same time (many examples included) • Can be set precisely to the exact second • Digital I/O port for advanced applications • Allows AppleWorks to time and date stamp all data automatically • Displays time and date on AppleWorks screens and files • Five-year warranty

Timemaster H.O. **$99.00**
BSR option (may be added later) **$29.00**

Order Timemaster today ... with 15-day money back guarantee and our "no hassle" five-year warranty. See your dealer or call (214) 241-6060, 9 a.m. to 11 p.m., 7 days, or send check or money order to Applied Engineering. MasterCard, Visa and C.O.D. welcome. Texas residents add 6¼% sales tax. Add $10.00 if outside U.S.A.

Applied Engineering
The Apple enhancement experts.

P. O. Box 798, Carrollton, TX 75006 (214) 241-6060

CP/M is a registered trademark of Digital Research, Inc.

CIRCLE **3** ON READER SERVICE CARD

UN CLASSIFIEDS

Members non-commercial rate .05/word
Members commercial rate .25/word
All others: 1.00/word. $25 minimum

Unclassified copy must be received 10 weeks prior to the 1st of the cover date month. For example, ad copy for the December issue should be in our hands by September 20th. A non-commercial rate only applies to single items for sale by a private party, such as a piece of used equipment. Send ad copy with check enclosed to the address below. Specify issue(s) in which to run.

Call -A.P.P.L.E.
Unclassified Advertisement Dept.
290 SW 43rd St.
Renton, WA 98055

RAMWORKS 1— With 512K including Applied Engineering Desktop Expander & Ramdisk software. $125 w/trade-in of extended 80–column card. No RGB expansion available. (408) 253-9238, Ninth Wave, 21907 Almaden Avenue, Cupertino, CA 95014.

LEARN CRAPS— on your apple with the "No Frills" CRAPS Simulator and tutor. Comes complete with Nevada craps rules and method of play. ONLY $19.95 VISA M/C orders. Call (800) 554-0017 Box 1012 or send cheque or money order to M. Coates, 150 NE 95th, Suite 301, Seattle, WA 98115.

WANTED— For elementary school and high school-Apple computers (II, IIe or IIc). Donations are tax deductible. Call John Henning (206) 824-6451, Kent.

EXTRAWORKS PRINTER UTILITY— Frustrated because AppleWorks won't let you access the full potential of your printer? Get color printing, mousetext, and nice half-height sub and super scripts from your Apple Image-Writer II (and slashed zeroes in AW 2.0), or italics and emphasized double-strike from Epson FX, RX, or MX/graftrax+. Works with above printers on AW 1.2, 1.3, or 2.0 by changing commands, leaving you the custom printer, and allowing the changes to work with proportional type. $20.00 ppd. (NJ residents include 6% tax) to: Eugene Whitehouse, 25 Kensington Avenue, #503, Jersey City, N.J. 07304.

RAM/EPROM BOARD— Use for program development, security applications or just instant access your favorite utilities. Plugs into any Apple II slot. User supplies memory chips. Board has 2 sockets to hold 2-2K RAM, 2-2K ROM or even 2-4K ROM for a maximum of 8K. Mix RAM/ROM! Soft switches control 2K bank select ($C800-CFFF). Hard switches control defaults. Only $39.00. RAK-WARE, 41 Ralph Road, West Orange, NJ 07052, (201) 325-1885.

INTEGRATED SOFTWARE— DO-RE-ME. Full-function Spreadsheet (38 built-in functions), Word Processor (35 embedded commands), DBMS (64 fields, reclength 2048, built-in functions in computed fields, multi-record display with vertical, horizontal Scroll, Zoom, Global update/delete), Auto - Report, 4 custom reports, MailMerge, DIF/ ASCII Import/Export, Context-sensitive help, Lotus 123 user interface, Desktop accessories, data integration using the Clipboard. Can read THE Spreadsheet, VisiCalc, MagiCalc, FlushCalc files directly (including formulas). supports multi-volume files, hard disk, up to Meg of memory. Runs on IIe, and IIc, under DOS 3.3 Unprotected, and very FAST. Probably the most sophisticated piece of software ever written for DOS 3.3. Introductory price: $49, plus $5 shipping. 30 day money-Back guarantee. Includes: System Disk, Sample Files disk, and a 380-page Manual. Free Tech Support. Write or call. Multisoft, 120 East 90th Street, box 5J, New York, NY 10128. (212) 534-4047.

$49 MIDI-MAGIC— Low cost one-way MIDI interface for the Apple IIc. No modifications required! Just plug into any music keyboard, synthesizer, organ or piano equipped with MIDI. Include cable (uses modem port) and 6-song disk. Large selection of QRS player-piano music available (write for catalog). Song disks also compatible with Passport MIDI card (or our own card: $99). RAK-WARE, 41 Ralph Road, West Orange, NJ 07052, (201) 325-1885.

INTELLIGENT DISASSEMBLER— Learn assembly language by studying programs written by the masters. DISASM converts machine code into clear readable assembler source code. A versatile utility program for experts and novices alike. Creates symbolic text files compatible with S-C, LISA Toolkit & other assemblers. Determine Page Zero usage instantly. Many features. Don Lancaster's ASSEMBLY COOKBOOK says DISASM is "absolutely essential". For Apple II, II+, IIe and Apple IIc STD 3.3 DOS (not copy-protected). Only $30.00 (or $50.00 w/commented source) prepaid. Write for catalog! RAK-WARE, 41 Ralph Road, West Orange, NJ 07052 (201) 325-1885.

FINALLY— An affordable detached keyboard — IBM style case for comfort, 1 key "CATALOG", PROGRAMMBLE, II+, IIe. Info/SASE. $45.00 up. Martek, POB 24, Novi, MI 48050

RENT APPLE II SOFTWARE— $5 monthly. No membership fee. Send large $.39 SASE for list. PS , 4229 Baker Ave. NW, Seattle WA 98107.

$2.65 BARD'S TALE MAP/HINT BOOKS— $2.65 postpaid. ASPEN SOFTWARE, 729 1/4 E. Portland St., Phoenix, Arizona 85006 (602) 253-0651.

75 CENT APPLE PUBLIC DOMAIN LIBRARY— Our 180 disk library contains thousands of programs for the Apple II, II+, IIe, IIc and IIGS: business, educational, games, EAMON adventures, music, graphics, utilities, hobbiest, more. Our copy-shipping fee is 75 cents/disk postpaid. We can copy onto disk(s) you mail to us for this fee or provide blank disks—$1.10 each extra. We can copy single-sided or double-sided. Our 42 page public domain catalog is $2. No additional charges or membership fees. ASPEN SOFTWARE, 729 1/4 E. Portland St., Phoenix, AZ 85006, (602) 253-0651.

PRINT SHOP USERS CLUB— We help you get the full benefit out of Print Shop! The $30.00 membership fee includes: 1200+ Graphics on 6 disks, plus a public domain utilities disk, plus a copy of the club newsletter, the Update Disk (articles and 100+ more graphics) Send $30.00 or a SASE for graphics list to: Print Shop Users Club, Box 216, Mercer Island, WA 98040.

TESTMAKER 2.1— for AppleWriter. Automatically compile and print any style test, answer key and questions-used list from one or more of your question files. Automatically renumbers test for multiple versions. Modify and renumber question files. Requires Applewriter. Cash, check or M.O. for $25.00 to Live Oak Software, P.O. Box 14814, Baton Rouge, LA 70898-4814.

THE WIZZ QUIZZER– Exciting new multiple quiz game with thousands of questions and responses to test your memory or use as a learning device. Keeps score and serenades you. Each quiz NEVER THE SAME the next time. (II+, c, e, gs). Quizzes: MOVIES, SPORTS, CARTOONS, INVENTIONS, ZODIAC, PLANETS, NAMES, FAMOUS INITIALS, etc. Send $8.50 ($8.93 - MD) to GARY HAYMAN, 8255 CANNING TERRACE, GREENBELT, MD 20770, and enjoy!

KERMIT— Why spend hundreds for communications software when the best is FREE! Just send me $8 for the disk, documentation, shipping and handling and I'll send you Columbia University's APPLE-II KERMIT, FREE"!". Jim Fox, BOX 14756, Pittsburgh, Pennsylvania, 15234.

WPL EXPANSION KIT— Big WPL program and glossary spaces, ten strings (four extra long), six big numerics, 32-bit divide, 16-bit mulltiply, eight new WPL options, peek & poke, clock interface (including IIGS), Get key option. Fast machine code modules. Kit $52 (includes $2 s/h). Specify AppleWriter 2.0, 2.1 MC/Visa. Guaranteed. Hotline support. Thompson & Thompson, 23072 Mullin Rd., El Toro, CA 92630 (714) 855-3838 PST.

APPLEWORKS TEMPLATES AND DATADISKS— Wide variety available! Prices start as low as $4.00 (plus shipping). Write for FREE catalog or send $2.50 for catalog and sample disk to WORDS FOR LESS, Box 212, Harrington Park, NJ 07640.

PRACTICAL BASIC ALGORITHMS— Stacks, queues, binary search and sort techniques, binary trees, database manager and much more all implemented in Applesoft. Contains source code for each plus and on-screen demonstration of data manipulation. An ideal preview for anyone planning to take a programming course. Also includes hi-res shape/font makers for graphics text and animation. Seven complete fonts (three are proportionally spaced) and many helpful hi-res hints. Extensive documentation. Apple II, 48K, DOS3.3. $19.95 + $3.00 S&H to: LambSoft Development, P.O. Box 42801, Philadelphia, PA 19101. (PA. and overseas add 6%).

CRYPTOMANIA!— 200 cryptogram puzzles on disk. Features puzzle and answer in different hi-res fonts, three difficulty levels, and save game. Create your own to play on the computer or print out for that traveling spouse. Apple II, 48K. All Lambsoft products are copyable and listable! Send $12.50 + $3.00 S&H to Lambsoft Development, P.O Box 42801, Philadelphia, PA 19101. (PA and overseas add 6%)

BEYOND PINBALL— The first Pinball Adventure Game—is not another computer simulation of pinball but stretches the concept as only a computer can. Contains not one but 16 games. Contains not one, but 48 screens to play on. Does not have two stationary button-controlled flippers, but instead, a joystick-controlled bumper and 360-degree-swinging bat movable all over the screens. Some of the types of obstacles you'll encounter: Black Holes, Forbidden Zones, Transport Tubes, Moving Objects and Switchable Objects. Includes extensive manual and game maps. All Apple II series. Total price = $21.95 ($35.00 for Pinball and Hi-Res Golf 3). Send check, M.O. or Visa-Mastercard number to Prime-Ware Creations, Box 1058, Philomath, OR 97370. Need more info? Send SASE. (503) 929-6221.

HI-RES COMPUTER GOLF 3— Eight Pro-Courses version. Now in its sixth year, this version contains the same quality features of prior versions and includes a beginners course and EIGHT professional tour golf courses. Contains true-to-scale fairways and greens (longer holes require multi-screens), green contours, sand traps, roughs, trees, water traps and variable wind conditions. Eight clubs in your golf bag. You physically swing the clubs or use the auto-swing feature. Bad swings may cause tops, undercuts, hooks, slices, etc. Up to a foursome can play. The package contains two diskettes, extensive manual and command card. All Apple II series. Total price = $21.95 ($35.00 for Golf and Beyond Pinball). Send check, M.O. or Visa/Mastercard number to Prime-Ware Creations, Box 1058, Philomath, OR 97370 (503) 929-6221.

INTERFACE CARDS FOR WHOLESALE PRICES— 16K RAM card $41.80, Z-80 card $39.60, 128K RAM card $71.70, 8088 card $64.10, 80 column card $50.90, Driver card $39.70, Color Graphic card $68.80, Monochrome graphic card $77.40, Mouse $75.30, Telex card $178.00, and 70 more different kinds of cards for special prices. Hunter electronic Co., LTD.P.O. Box 53-130, Taipei Taiwan, Rep. of China.

US $8.00 including disk! Thousands of famous Business, Education, games, and programs for Apple II, IIe, IIc and IBM /PC. Details US $1.00 Reliant, PO Box 33610 Sheungwan Hongkong.

Preferred Computing

ORDER TOLL FREE
1-800-327-7234

Computer Accessories — When Only the Best Will Do!

VISA COD MasterCard

We only sell proven products that work. By selling only the industry best, we can assure you of the highest level of reliability and performance. Selling large volumes of a smaller product line allows us to have the lowest discount prices and fast same day shipment.

AE APPLIED ENGINEERING™

RamWorks III™ If you have a IIe this card is perfect and if you use AppleWorks, it's an absolute must! Applied Engineering was the first company to expand AppleWorks and they're always 6-12 months ahead of the competition. And you can stay current with our FREE SOFTWARE UPGRADES. Why settle for a partial compatible when you can have a real RamWorks at a discount price.

RamWorks III 256K **$169**
RamWorks III 512K **$209**
RamWorks III 1 MEG **$269**
RamWorks III 1.5 MEG **$389**

65C816 16 Bit Card Requires RamWorks II or III and Apple IIe. **$149**

RamWorks RGB Option Even if you bought a RamWorks just to use the RGB option, it would be worth it. Runs both Apple and IBM RGB monitors, and is compatible with Apple's standard for RGB, includes cables. **$99**
(See our new RGB monitor ar right.)

TransWarp™ We tested all the accelerators and TransWarp was 20-300% faster than the others. Runs cool too! **$219**

Z-80 Plus™ Runs all CP/M® software. Includes operating system and manual, a great value. **$109**

Timemaster H.O.™ Timemaster H.O. has the most features and is the easiest to use. It comes with more software than any other clock card. It even adds 15 new commands to BASIC! **$79**

Phasor Simply the best sound and speech synthesizer for the II+, IIe, or IIgs. **$139**

SerialPro A powerful multifunction card with serial ports and clock/calendar **$129**

Viewmaster 80™ Super resolution, total software compatibility and everything you need to run AppleWorks on a II+ or Franklin. **$109**

RamFactor™ If you have a II+ or would rather use slots 1-7 in your IIe or IIgs, RamFactor is for you. It is fully compatible with Apple's memory card, runs AppleWorks on a IIe, II+ or IIgs and has a battery back-up option.

RamFactor 256K **$189**
RamFactor 512K **$224**
RamFactor 1 MEG **$299**
Battery Back-up **$149**

Extended 80 Column Card with 64K Comes with full documentation. **$69**

gsRAM & gsRAM Plus These cards offer much greater performance and expansion capabilities than Apple's memory card and unlike most others, gsRAM and gsRAM Plus use standard RAM's so further expansion is easy and economical. Only gsRAM and gsRAM Plus can expand AppleWorks 2.0 internal limits and include a built-in printer buffer.

gsRAM 256K **$139**
gsRAM 512K **$179**
gsRAM 1 MEG **$249**
gsRAM 1.5 MEG **$319**
gsRAM Plus 1 MEG **$529**

IIc System Clock™ **$49**

Z-RAM Ultra — For all IIc's

Ultra 1: Memory Only	256K	**$169**
	512K	**$209**
Ultra 2: Memory & Clock	256K	**$209**
	512K	**$249**
	1 MEG	**$319**
Ultra 3: Memory, Clock & CP/M	256K	**$259**
	512K	**$299**
	1 MEG	**$369**

More Quality Products

Ultra Hi-Res RGB Monitor/TV This Sony RGB monitor uses a 13" ultra hi-res CRT for super sharp 80 column text and great graphics. But in addition to being the best RGB monitor available this unit is also a 180 channel cable-ready television set with remote control. Try it. You WILL like it. (List $625) Only **$499**

RGB Cable For above **$29.00**

5-1/4" Disk Drive Super quiet direct drive compatible with all Apple software and Apple controller card. A top quality unit with 1 year warranty. Only **$119**

5-1/4" Disk Controller Runs up to 2 drives. **$49**

16K RAM Card For Apple II and II+ **$69**

Heavy Duty Apple Power Supply With over twice the output current of Apple's original power supply. **$59**

Kensington System Saver Fan with 2 power outlets and surge protector. **$69**

Internal 1200 Baud Modem Cermetek 1200, for Apple IIe, II+ and IIgs runs at 1200 and 300 baud. Auto-dial, auto-answer with built-in speaker. Hayes compatible, with built-in communications software. Comes complete with cable and manual. (List $299) **$149**

Pinpoint's AppleWorks Accessories The best available and they are written to fully utilize Applied Engineering's RAM cards. (Includes RAM enhancement software). Together a $118 value. Only **$69**

MacroWorks and Super MacroWorks ". . . the best single AppleWorks add-on available today." A+ Magazine. Add new word processing power, create your own Macros and save lots of keystrokes. Use mouse with AppleWorks. No more pathnames! Just select from a menu. Works with Applied Engineering cards and software. Buy MacroWorks for AppleWorks versions 1.0 - 1.3. Buy Super MacroWorks for AppleWorks version 2.0.

MacroWorks (List 34.95) **$29.00**
Super MacroWorks (List 49.95)... **$39.00**

256K RAMS 150ns. 5 year warranty. Set of 8 **$39**

VIP Professional Lotus 1-2-3 Workalike. For IIe & IIc (List $249) **$169**
For IIgs (List $299) **$199**

RamWorks™ That's right, RamWorks not RamWorks II. RamWorks is the memory card that started the revolution. RamWorks can expand to 512K on the main board and up to 1 meg with an optional 512K piggyback and can accept Applied Engineering's new RGB option. A 512K RamWorks gives AppleWorks a 360K desktop. Comes with Applied Engineering's newest software. With 512K **$139**

Why buy from us?

1. Toll free 800 numbers for orders.
2. We almost always ship from OUR stock. Call us before 3 P.M. and it ships today.
3. All memory cards have factory installed and tested RAM chips.
4. We're nice guys that want to help. We want our business to grow because of repeat customers and word of mouth.
5. We handle only the industry's best. We've tested and used all the products we sell.
6. 15 day money back satisfaction guarantee on hardware. If you don't like something, we'll give you your money back. No hassles.
7. We'll support, service and warranty everything we sell.
8. No surcharge is added for charge cards and we won't charge your card until we ship.
9. No sales tax is collected on orders outside Texas.
10. All shipments are insured by us.

Order by mail: Send check or money order to Preferred Computing, P.O. Box 815828, Dallas, TX 75381
Order by phone: 1-800-327-7234 Visa, MasterCard or COD
Texas, Hawaii and Alaska 1-214-484-5464

Orders only call 1-800-327-7234
Questions and Customer Service call 1-214-484-5464
Texas residents add 6¼% sales tax
Prices subject to change without notice

Manufacturing Company, Inc.

The Experts in Accessories

CURTIS

DIAMOND SP-1
Switched; adapts to duplex outlet; • 6 Outlets, 15 Amps Total 125 Volts, 1875 Watts • Operating Temperature: -40°C to + 85°C • Peak Pulse Power Dissipation @ 25°C, 585,000W for 20 Micro Sec. • Maximum Clamping Voltage @ Test Current 50A (8/20 Micro Sec.) 340V Clamping Occurs Line to Line (Normal Mode) and Line to Ground (Common Mode) • Clamping Response Time: 1 Nanosecond (1 × 10–9 sec.) • Dielectric Test: 3000 VAC 60 Sec. • Forward Surge Rating @ 25°C, 1440 Amps for 1/120 Sec., Lifetime Warranty
Retail $49.95

RUBY SPF-2 Switched; EMI-RFI filtered; plugs into duplex outlet; 6 ft. power cord, Built-in Circuit Breaker • 6 Outlets, 15 Amps Total 125 Volts, 1875 Watts • Operating Temperature: -40°C to + 85°C Peak Pulse Power Dissipation @ 25°C, 585,000W for 20 Micro Sec. • Maximum Clamping Voltage @ Test Current 50A (8/20 Micro Sec.) 340V Clamping Occurs Line to Line (Normal Mode) and Line to Ground (Common Mode) • Clamping Response Time: 1 Nanosecond (1 × 10–9 sec.) • Dielectric Test: 3000 VAC 60 Sec. • Forward Surge Rating @ 25°C, 1440 Amps for 1/120 Sec. • Mode Noise Protection: Normal and Common Mode • Noise Rejection Frequency: 150KHZ–30MHZ • Attenuation: 5DB–58DB, Lifetime Warranty
Retail $89.95

EMERALD SP-2 Switched; plugs into duplex outlet; 6 ft. power cord, Built-in Circuit Breaker, 6 Outlets, 15 Amps Total 125 Volts, 1875 Watts From Single or All Combined • Operating Temperature: -40°C to + 85°C • Peak Pulse Power Dissipation @ 25°C, 585,000W for 20 Micro Sec. • Maximum Clamping Voltage @ Text Current 50A (8/20 Micro Sec.) 340V Clamping Occurs Line to Line (Normal Mode) and Line to Ground (Common Mode) • Clamping Response Time: 1 Nanosecond (1 × 10–9 sec.) • Dielectric Test: 3000 VAC 60 Sec. • Forward Surge Rating @ 25°C, 1440 Amps for 1/120 Sec., Lifetime Warranty
Retail $59.95

SAPPHIRE SPF-1
Switched, EMI-RFI Filtered, adapts to duplex outlet; • 3 Outlets, 15 Amps Total 125 Volts, 1875 From Single or All Combined • Operating Temperature: -40°C to + 85°C • Peak Pulse Power Dissipation @ 25°C, 585,000W for 20 Micro Sec. • Maximum Clamping Voltage @ Test Current 50A (8/20 Micro Sec.) 340V Clamping Occurs Line to Line (Normal Mode) and Line to Ground (Common Mode) • Clamping Response Time 1 Nanosecond (1 × 10–9 sec.) • Dielectric Test: 3000 VAC 60 Sec. • Forward Surge Rating @ 25°C, 1440 Amps for 1/120 Sec. • Mode Noise Protection: Normal and Common Mode • Noise Rejection Frequency: 150KHZ–30MHZ • Attenuation: 5DB–58DB, Lifetime Warranty
Retail $79.95

All Curtis Surge Protectors meet or exceed IEEE standards 587-1980 Guide for Surge Voltages in AC Power Circuits.

The Ones To Ask For

Curtis Electronic Surge Protectors are built to provide surge protection for any microcomputer for a lifetime. Constructed with precision, using advanced semiconductor technology, the Curtis Jewels prevent damage to microcomputer systems and data from powerline overvoltage

"surges, spikes and glitches." Surge protection from the Curtis Jewels is your guarantee of the best line of defense for your valuable hardware and data, at a price that won't shock you. Get your computer the protection it deserves with Curtis — the experts in accessories.

Curtis products are available nationally from leading Dealers, Distributors, and Retail Chains.
In Canada: Micro-Computer Products, P.O. Box 235, Ajax, ONT, Canada L1S 3C3, (416) 427-6612

For the Curtis dealer nearest you call (603) 924-3823

Manufacturing Company, Inc.

CURTIS

305 Union Street, Peterborough, NH 03458

C-1

CIRCLE **5** ON READER SERVICE CARD